Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

**Document**
D-91-09

# Proceedings

# MLNLO'91

## Machine Learning of Natural Language and Ontology

David Powers and Larry Reeker (Eds)

March 1991

**Deutsches Forschungszentrum für Künstliche Intelligenz GmbH**

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

# Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies ADV/Orga, AEG, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Mannesmann-Kienzle, Philips, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- Intelligent Engineering Systems
- Intelligent User Interfaces
- Intelligent Communication Networks
- Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

# MLNLO'91 – Machine Learning of Natural Language and Ontology

**David Powers & Larry Reeker (Eds)**

DFKI-D-91-09

Originally distributed as Working Notes of the AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, March 26-28, 1991, Stanford University.

© Copyright 1991

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following notice.

These controbitutions to the Working Notes of the AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, March 26-28, Stanford University, are copyright by the individual authors, and are reproduced here with the permission of the authors and the American Association for Artificial Intelligence.

# Preface

People begin their lives without the ability to speak any natural language and are able, in a few short years, to develop a linguistic competence that enables them to function as a writer, scholar, politician -- whatever they choose to become. They may, in fact, learn to communicate in several natural languages. These remarkable phenomena of language learning have amazed most of us at one time or another, and it is only natural that we have tried to use computers to study or even duplicate them — with only partial success to report at this date.

The AAAI Spring Symposium on Machine Learning of Natural Language and Ontology (MLNLO) provided an opportunity to get together and discuss the partial successes and the research challenges that lie ahead. It was a rare opportunity, because the work has tended to be reported in fragments, a thesis here or there, a paper at an AI or computational linguistics conference, another at a psychology or linguistics or child language conference or in a philosophy journal. The field is naturally highly multi-disciplinary, and the interested researchers all speak their own languages — not just natural languages, but specialized disciplinary dialects, laden with the theoretical constructs and assumptions of each discipline. So the MLNLO Symposium provided a forum for useful interchange of ideas.

"Learning of natural language" is a simple-sounding term that covers a number of phenomena. On the one hand, there are various aspects of language to be learned, such as the sounds significant in a particular language (phonology), words (lexicon) and their variations (morphology), the structure of meaningful utterances (syntax), meaning and its relation to the lexicon and to syntactic structure (semantics). On the other hand, there are the different components of learning: inducing the data to be learned from raw lingustic and non-linguistic data, somehow codifying those data into an internalized, structured system that can be used in an automatic manner, generalizing to be able to deal with new inputs never heard before and produce new outputs never uttered before. The learning of ontology, the understanding of what exists in the world, is closely linked with the learning of language.

At the symposium, 50 participants discussed contributions in all the areas mentioned — with 20 full length presentations and a similar number of "advertizing" spots which allowed virtually all groups some air-time. The first paper, a technical preface to the working notes as distributed at the symposium, will provide a good overview of the field and the volume. It was initially written as a background paper to the call for submissions, and was subsequently reworked to provide both background to the symposium and a summary of how the various contributions to the volume fit into the field of Machine Learning of Natural Language and Ontology.

At the end of the symposium, participants took time out to review the value of the symposium and look toward the future. It was resolved that we instigate a regular program of MLNLO events, a newsletter, resource sharing (software, texts, etc.), and further symposia, workshops and conferences. The symposium participants also felt that the "Working Notes" of the MLNLO Symposium were a landmark volume worthy of further distribution — a resolution which led directly to its publication in this form.

The editors wish to thank the German AI Institute (DFKI) for their support during the organization of the symposium, and in particular for the publication of this proceedings in their research report series.

David Powers, DFKI,
University of Kaiserslautern
Germany

powers@dfki.uni-kl.de

Larry Reeker, C & SE Divn,
Institute of Defense Analyses
Washington

reeker@cs.ida.org

# MLNLO'91

# Machine Learning of Natural Language and Ontology

David Powers
and
Larry Reeker
(Editors)

Working Notes of the AAAI Spring Symposium, March 26-28 1991, Stanford University.

These contributions are copyright by the individual authors, and are reproduced here with the permission of the authors and the American Association for Artificial Intelligence.

# American Association for Artificial Intelligence
# Spring Symposium Series 1991

# MACHINE LEARNING OF NATURAL LANGUAGE & ONTOLOGY

**Jordan Hall, Room 040**

## TUESDAY 26th March 1991

9:00 **COGNITIVE SCIENCE** – *Psycholinguistics* (Powers)

**David Powers** (short)
*WELCOME*

**Mallory Selfridge** (long)
*How Do Children Learn to Recognize Ungrammatical Sentences*

**Steven Lytinen & Carol Moon** (long)
*Cognitive Modelling of Second Language Acquisition*

**James Martin** (short)
*Learning Conventional Metaphors and Learning Using Conventional Metaphors*

**Neza van der Leeuw** (short)
*A Data-Driven Model of First Language Acquisition*

11:00 **COMPLEXITY THEORY** – *Learnability* (Berwick)

**Janet Fodor** (long)
*Making Phrase Structure Grammars learnable*

**Sanjay Jain & Arun Sharma** (long)
*Restrictions on grammar size in language identification*

**Leona Fass** (short)
*Applying Some CFL Learnability Results to Natural Languae Learning*

14:00 **TRADITIONAL APPROACHES** – *Machine Learning* (Feldman)

**Pat Langley** (long)
*Machine Learning and Language Acquisition*

**Robin Clark** (long)
*A Computational Model of Parameter Setting*

**Robert Berwick** (short)
*Parsing and Language Acquisition: From rules to parameters*

16:00 **TRADITIONAL APPROACHES** – *Explanation-Based Learning* (Reeker)

**Scott Stethem** (long)
*Explanation-Based Learning from Rule-Governed Features in Phonological Representations*

**Christer Samuelsson & Manny Rayner** (long)
*Quantitative Evaluation of the Utility of Explanation-Based Learning as a Tuning Tool for Large-Scale Natural Language Interfaces*

18:00 **RECEPTION** – *Tresidder Oak Lounge*

# WEDNESDAY 27th March 1991

9:00 **SYMBOL GROUNDING** – *Problem and Practice* (Powers)

**Stevan Harnad** (long)
*The Symbol Grounding Problem and Categorial Perception*

**Jeffrey Siskind** (long)
*Naive Physics, Event Perception, Lexical Semantics and Language Acquisition*

**Brian Bartell & Garrison Cottrell** (short)
*A Model of Symbol Grounding in a Temporal Linguistic Environment*

**Susan Weber** (short)
*Miniature Language Acquisition and the L0 Project*

11:00 **SYMBOL GROUNDING** – *Symbol and Semantics* (Harnad)

**Vasant Honavar** (long)
*Towards Computational Models of Natural Language Acquisition*

**Uri Zernik** (long)
*Learning from Authentic Corpus*

**Stefan Wermter** (short)
*Hybrid Symbolic/Connectionist Methods for Natural Language Processing*

**Jan Scholtes** (short)
*Learning Simple Semantics by Self-Organization*

14:00 **TRADITIONAL APPROACHES** – *Semantics and Phonology* (Lehnert)

**Peter Hastings & Steven Lytinen** (long)
*Automatic Acquistion of Word Meanings*

**Michael Brent** (short)
*Automatic Semantic Classification of Verbs*

**Narciso Jaramillo & Marti Hearst** (short)
*Acquiring the Semantics of Simple Phrasal Patterns Using COBUILD*

**Jeffrey Siskind** (short)
*Acquiring Core Meanings of Words, Represented as Jackendoff-style Conceptual Structures, from Correlated Streams of Linguistic and Non-Linguistic Input*

**Mark Ellison** (short)
*Discovering Planar Segregation*

16:00 **TRADITIONAL APPROACHES** – *Syntax and Structure* (Marcus)

**Mitchell Marcus** (short)
*Deducing Linguistic Structure from Large Corpora*

**David Leblanc & Henry Davis** (short)
*A Model of the Development of Phrase-Structure*

**David Magerman** (short)
*Mutual Information*

**Rick Kazman** (short)
*On Building a Model of Grammar from Information in the Lexicon*

**Deborah Dahl** (short)
*Applications of Training Data in Semantic Processing*

19:30 **PLENARY** – *Kresge Auditorium* (Patel-Schneider)

# THURSDAY 28th March 1991

9:00 **SYSTEM DEVELOPMENT** – *Computing and Applications* (Selfridge)

**Larry Reeker** (short)
*Language Learning and Adaptive User Interfacs*

**Claire Cardie & Wendy Lehnert** (short)
*Learning Complex Syntax within a Semantic Parser*

**Marc Goodman** (short)
*A Case-Based, Inductive Architecture for Natural Language Processing*

**Bill Hart** (short)
*Recurrent Neural Nets for Natural Language Acquistion*

11:00 **JOINT PANEL** – *Connectionist Learning of Natural Language* (Powers & Dolan)

**Jordan Pollack** (short)
*Induction as Phase Transition*

**Charles Dolan** (short)
*Why Natural Language Processing needs Connectionism.*

**David Powers** (short)
*How Far Can Self-Organization Go? Results in unsupervised language learning*

**Jane Hill** (short)
*Hybrid Models of Language Learning*

**Andreas Stolcke** (short)
*Vector Space Grammars and the Acquisition of Syntactic Categories: What connectionist and traditional models can learn from each other*

12:30 **END** – *Formal End of Symposium*

## DISCUSSION

The aim of the MLNLO symposium is to encourage interaction and promote discussion amongst Language and Learning researchers. With this in mind, in addition to the usual long talks, we have included a similar number of short spots which allow people to introduce themselves, their work and their groups. The long talks are a nominal 30 mins and the spots 10 mins.

These times include a few minutes for questions and discussion, as usual, but additional time is allowed at the end of each session for general discussion. As this discussion is not intended to be limited to the current session, but may allow picking up and relating of earlier themes, increasing amounts of discussion time are allowed as the day wears on, and as the days roll by.

As we are not running parallel session, chairmen also have the freedom to allow discussion to continue following a particularly provocative presentation, taking into account the additional discussion time in that session. For this reason precise times for talks are not shown in this programme.

# Table of Contents



NOTES:

1. The first page of some of the papers contains a background sketch of the participant(s). For some participants, only the sketch is included.

2. Due to a possible conflict of commitments, Uri Zernik may not be able to attend or provide a paper. If he is able to do so, the paper will be available at the Symposium as an addendum to this volume.

# Preface:

## Goals, Issues and Directions in Machine Learning of Natural Language and Ontology

**David M. W. Powers, FB Informatik**
University of Kaiserslautern, FRG
powers@informatik.uni-kl.de

### 1. INTRODUCTION

This is it! The AAAI Spring Symposium on Machine Learning of Natural Language and Ontology (MLNLO) has become a reality, and this volume of "Working Notes" provides an almost exhaustive overview of current work in this area. This is the first real opportunity for researchers from all disciplines and all countries to come together and explore the relationships between Learning (Human and Machine) and Natural Language. We not only have input from researchers in Computer Science and Artificial Intelligence (Machine Learning, Natural Language, Vision, Neural Nets, Parallelism) but contributions from other fields (Linguistics, Psycholinguistics, Philosophy).

This Preface seeks to provide a brief guide to the contributions, drawing attention to individual contributions in the context of a review of the field. The content overlaps to a large degree that of [Powe91], but contains material particular to this symposium.

The symposium committee hopes that you will enjoy reading these contributions and participating in the symposium, and trust that you will be as impressed with the progress represented here as we were.

#### 1.1 Committee

David Powers, Manny Rayner, Larry Reeker, Chris Turk.

#### 1.2 Reference

[Powe91] David M. W. Powers, "Goals, Issues and Directions in Machine Learning of Natural Language and Ontology", SIGART Bulletin, 2, #1, January 1991. Also available as SEKI Report SR-90-14, University of Kaiserslautern FRG.

### 2.1 Applicability of traditional machine learning.

#### 2.1.1 Introduction

Under the heading of Machine Learning, we particularly have in mind work in concept learning - clearly related to semantics and potentially to syntax and pragmatics. We are also interested in the role of teacher and critic, including automatic generation of examples, implicit criticism, unsupervised learning etc. Application of traditional techniques to facets of language are fundamental in that they are immediately accessible and connect with a considerable body of previous work.

#### 2.1.2 Bibliography

Angluin, Dana and Carl H. Smith, "Inductive Inference: Theories and Methods," Computing Surveys, vol. 15, no. 3, pp. 238-269, September 1983.

DeJong, G. and Mooney, R. "Explanation-Based Learning: An Alternative View" Machine Learning vol. 1, pp145-176, 1986. Fisher, D. H., "Knowledge Acquisition via Incremental Conceptual Clustering," Machine Learning, vol. 2, pp. 139-172, 1987.

Forsyth, R. and R. Rada, Machine Learning: Applications in Expert Systems and Information Retrieval, Ellis Horwood, Chichester, 1986.

Haussler, D., "Learning conjunctive concepts in structural domains," Machine Learning, vol. 4, pp. 7-40, 1989.

Helmbold, D., R. Sloan, and M. K. Warmuth, "Learning nested differences of intersection-closed concept classes," Machine Learning, vol 5. pp. 165-196, 1990. Also available as UCSC-CRL-8919, Comp. Res. Lab., Univ. California Santa Cruz, 1989.

Hunt, E. B., J. Marin, and P. J. Stone, Experiments in induction, Academic Press, New York NY.

Laird, J. E., P. S. Rosenbloom, and A. Newell, "Chunking in SOAR: The Anatomy of a General Learning Mechanism," Machine Learning, vol. 1, pp. 11-46, 1986.

Langley, P., "Learning search strategies through discrimination," Int'l Jnl of Man-Machine Studies, vol. 18, pp. 513-541, 1983.

Lenat, D. B., "EURISKO: A Program That Learns New Heuristics and Domain Concepts; The Nature of Heuristics and Domain Concepts," Artificial Intelligence, vol. 21, no. 1, pp. 61-99, 1983.

Michalski, R. S., I. Mozetic, J. Hong, and N. Lavrac, "The multi-purpose incremental learning system AQ15 and its testing application in three medical domains.," Proc. AAAI-86, Philadelphia PA, 1986.

Mitchell, T. M., Keller, R. M. and Kedar-Cabelli, S. T., "Explanation-Based Generalization: A Unifying View" Machine Learning, vol. 1, pp47-80.

Muggleton, S. and W. Buntine, "Machine invention of first-order predicates by inverting resolution," Proc. 5th Int'l Conf. on Machine Learning, pp. 339-352, Morgan Kauffman, San Mateo CA, 1988.

Quinlan, J. R., "Induction of decision trees," Machine Learning, vol. 1, pp. 81-106, 1986.

Riesbeck, Christopher K., "Failure-driven Reminding for Incremental Learning," 7th International Joint Conference on Artificial Intelligence, pp. 115-120, 1981.

Samuel, A. L., "Some studies in machine learning using the game of checkers II - recent progress," IBM Jour. R & D, vol. 11, no. 6, pp. 601-617, 1967.

Sammut, Claude and R. Banerji, "Learning concepts by asking questions," in Machine Learning: an Artificial Intelligence Approach, ed. R. S. Michalski, J. G. Carbonell and T. M. Mitchell, vol. 2, 1986.

Shapiro, E., "A general incremental algorithm that infers theories from facts," Proc. 7th IJCAI, pp. 446-451, 1981.

Winston, P. H., "Learning structural descriptions from examples," in The Psychology of Computer Vision, McGraw-Hill, 1975.

#### 2.1.3 Significance

We here pick out some of the above work for particular comment, singling out that which has been particularly influential and crudely indicating streams of development.

Angluin's work is highly regarded itself (see also section 2), and the review presented here is a good place to start for a survey of inductive methods.

Samuel's checker playing programs is one of the first major success stories of machine learning, and indeed the signature table technique can be said to be a precursor of both today's neural network tradition (see section 5) and the statistical approaches represented by the line of Hunt, Michalski and Quinlan, which has become particularly influential for Knowledge Engineering purposes (Automatic Acquisition of rules for Expert Systems). To the extent that language is regarded as rule based, there is an obvious potential for application of these techniques of rule learning, and in particular classification. Such techniques have been used in MLNL (see section 7).

Forsyth and Rada is a reasonable text, particularly in relation to this type of learning, but also in relation to evolutionary learning. On this point, it may be noted that there are criticisms that language cannot be learnt (see section 2) but that language behaviour is selected from an evolved capacity for language. These can in part be answered by pointing out that we could actually employ, for Machine Learning, any "techniques" used by such evolution - although we may not be happy with the time scale!

The work of Winston and the line of Banerji, Cohen and Sammut on developing logical representations of concepts, is particularly interesting for its showing that the role of teacher may be separated from that of critic. In Sammut's system, after a generalization step the system provides its own new example to test the validity of the generalization, and only requires positive or negative criticism. The teacher need only provide the initial (positive) example. The critic must provide feedback on every example. This type of approach is particularly appropriate for learning of semantics. It is primarily in a neural network or statistical context that I am aware of inductive learning applications where criticism is not used (see sections 4 & 5).

But there are types of learning other than induction, the learning of new concepts or rules. There is also learning to do things better or faster. Explanation-based learning (Mitchell et al., DeJong and Mooney), the version space technique (Mitchell), EURISKO (Lenat) and Chunking (Laird et al.) have also their applications to

MLNL.

We note that Lenat's more recent work on CYC, which uses explicit acquisition rather than machine learning in the present stage of the project, deals with other problems related to MLNLO and is referenced in section 6. Beckwith et al's work with Miller (see section 4) is in some ways similar, concentrating on different directions than MLNL at the moment in the application of psycholinguistic results.

A final classification of learning systems can be made on the basis of whether they are capable of incremental learning or not. Winston, Fisher, Shapiro and Riesbeck particularly address this "problem". Some of the above techniques like to work with full information, or a sample, others are inherently incremental. Some find restricting themselves to incremental learning a disadvantage. However, given that Natural Language can be learnt with incremental exposure it could well be that incremental algorithms can be more efficient for a class of problems which includes MLNL (see section 3).

### 2.1.4 In this volume

In this volume, Pat Langley provides a further review of the applicability of Machine Learning techniques to Natural Language. Scott Stethem and Christer Samuelsson with Manny Rayner apply Explanation Based Learning to completely different domains - Phonology and Parser Tuning!

Robin Clark presents Genetic Learning techniques which Berwick has started using for parameter selection for a particular parsing model.

## 2.2 Applicability of traditional linguistics and parsing techniques.

### 2.2.1 Introduction

Some MLNL approaches are based on traditional theories from linguistics and elsewhere. Learnability provides a very practical test for a linguistic theory. A good approach to parsing should relate to a good approach to learning syntax. Many approaches however are based on non-linguistic traditions, notably neural nets. It is especially important to consider the connections between different disciplinary approaches.

### 2.2.2 Bibliography


Catania, A. C. and S. Harnad, The Selection of Behavior. The Operant Behaviorism of B. F. Skinner: Comments and Consequences., Cambridge University Press, New York NY, 1988.

Chomsky, Noam, Aspects of the Theory of Syntax, MIT Press, Cambridge MA, 1965.

Derwing, Bruce L., Transformational Grammar as a Theory of Language Acquisition, Cambridge University Press, Cambridge UK, 1973.

Halliday, M. A. K., "Language Structure and Language Function," in New Horizons in Linguistics, ed. J. Lyons, Penguin, Harmondsworth, Middlesex UK, 1970.

Halliday, M. A. K. and R. Hasan, Cohesion in English, Longman, London UK, 1976.

Jackendoff, Ray, Semantics and Cognition, MIT Press, Cambridge MA, 1983.

Kay, M., "Parsing in Unification Grammar," in Natural Language Parsing, ed. Dowty, Karttunen and Zwicky, 1985.

Marcus, M., A Theory of Syntactic Recognition for Natural Language, MIT Press, Cambridge MA, 1980.

Pereira, Fernando C. N. and David H. D. Warren, "Definite clause grammars for language analysis - a survey of the formalism and a comparison with augmented transition networks," Artificial Intelligence, vol. 13, no. 3, pp. 231-278, 1980.

Pike, Kenneth L., Phonemics, Summer Institute of Linguistics, Santa Ana CA, 1947.

Pike, Kenneth L., Language in Relation to a Unified Theory of the Structure of Human Behavior, Mouton, The Hague, Holland, 1954/1967.

Pike, Kenneth L. and E. G. Pike, Grammatical Analysis, Summer Institute of Linguistics (and University of Texas at Arlington), Dallas, Texas, 1977.

Popper, K. R., The Logic of Scientific Discovery, Hutchinson, London UK, 1959.

Schank, Roger C., "Conceptual Dependency: A Theory of Natural Language Understanding," Cognitive Psychology, vol. 3, no. 4, pp. 552-631, 1972.

Schank, Roger C., Conceptual Information Processing, North Holland, 1975.

Schubert, L. K., "Problems with Parts," 6th International Joint Conference on AI, pp. 778-784, 1979.

Skinner, B. F., Verbal Behaviour, Appleton-Century-Crofts, New York NY, 1957.

Skinner, B. F., "The Phylogeny and Ontogeny of Behaviour," in Contemporary Issues in Developmental Psychology, ed. E. Endler, L. Boulter, and H. Osser, pp. 62-77, Holt, Rhinehart and Winston, New York, 1968. Reprinted from Science, 1966, Vol 153, pp 1205-1213.

Vanderslice, R., "The Prosodic Component: Lacuna in Transformational Theory," P-3874, Rand Corporation, Santa Monica CA, November 1968.

Woods, W. A., "Transition Network Grammars for Natural Language Analysis," CACM, vol. 13, pp. 591-606, 1970.


### 2.2.3 Significance

The above references have been advertized as traditional linguistics and parsing techniques. Some therefore require a word of explanation on their inclusion!

The behaviourist references, to Skinner and critiques of his work, are included here because of his critiques of linguistics, not to mention the confrontation between his approach and Chomsky's (which I haven't documented here). The reference to Popper's philosophy of science doesn't really belong under Machine Learning either, but it is fundamental to some of the issues in Linguistics today, and it is also relevant to Machine Learning of Ontology - in the sense that Science is the process by which, as a society, we learn about our world.

Some of the AI work, Schubert's for example, is quite a way from parsing, but deals with issues important to semantics, and is part of the heritage we have when we come to do MLNLO. Other work, Schank's and Wood's, are particularly fundamental traditions in NL. Schank is concerned also with semantics, and conceptual dependency theory is one of the most well developed semantic representations. The work of his group moreover extends to MLNL projects (see section 7). Pereira and Warren are particularly important as representatives of the Logic Programming approach to NLL, and learning techniques have also been applied to their work (see section 7).

Pike is represented for his broad view of language and behaviour, being one of the first to recognize that language and ontology cannot be separated. His theory and methodology of Phonology are still standard, and his generalization to the Tagmemics theory of grammar is significant for its supporting of phrase structure with cohesion, and has also proven the base for some MLNLO work (see section 7). Halliday is responsible for the Systemic grammatical theory brought to the attention of the AI world by Winograd. He also emphasises the role of cohesion.

But even after 30 years, Chomsky's school of Transformational Generative Grammar (TGG) remains dominant. It has, moreover, had a significant influence on Psycholinguistics (see section 4), which in turn has generated criticism of TGG, represented here by Derwing and Vanderslice. It too has been used as a guide for MLNL work, and TGG has benefited from criticism from this source as well (see section 7). The current manifestations in Government and Binding Theory highlight fundamental and apparently universal linguistic properties which should be predictive points on any MLNL modelling agenda.

2.3 Goals and Issues

GOAL: Neural investigations need to determine and characterize the nature and role of the human (animal) wetware, as well as stretching the limits of neural inspired models.

ISSUE: What are the limits of genetic determination, boundary conditions and self?

2.3.1 In this volume

In this volume, Robert Berwick discusses the parameterized parser to which he is applying genetic learning techniques for parameter setting, whilst Janet Fodor discusses modifications to phrase structure grammar (GPSG and HPSG) to achieve better learnability.

David LeBlanc and Henry Davis look language acquisition in terms of a modified Government and Binding Theory. Rick Kazman and Deborah Dahl pursue more probabilistic and lexicon centred approaches. Mitch Marcus and David Magerman are looking at combining aspects of TGG with the competitive structuralist approach, reporting encouraging results even in the absence of an explicit grammar.

The missing component in much traditional Linguistics, semantics, is addressed in related fashion by Zernik, Honavar, Wermter and Scholtes. In some of this work, interestingly, the distinction between syntax and semantics starts to become rather fuzzy!

Further exploration of the last observation brings us back to the O in MLNLO, and the Symbol Grounding Problem of Section 6. And the whole question of learnability in the different models, already touched on here, brings us to Complexity Theory, which we will look at now.

## 3. COMPLEXITY THEORY

### 3.1 Formal results on learning and language constraints.

3.1.1 Introduction

Results and proposals based on complexity theory have been driving forces in some schools of linguistics and psycholingustics - notably the contributions of Gold and Chomsky. New approaches, algorithms and claims need to be considered in the light of such results, and appropriate new analyses should be developed.

Rigourous mathematical analysis is an important source of criticism for Cognitive Science research. Publication of results can shape the whole future of a field, firmly closing off former paths of attack, and opening up others. Unfortunately, the effect has not always been positive. In some noteworthy cases, the wider Cognitive Science community has taken a result at face value, applied it far outside the applicable conditions (spelled out by the original author), and interpreted it without common-sense reflection on and reinterpretation of the natural world correlates of the analyzed system. This list includes a number of such examples. It pays to consider these results first hand!

#### 3.1.2 Bibliography


Angluin, Dana, "Inference of reversible languages," J. ACM, vol. 29, pp. 741-765, 1990.

Angluin, Dana, "Negative Results for Equivalence Queries," Machine Learning, vol. 5, pp. 121-150, 1990.

Board, Raymond and Leonard Pitt, "On the Necessity of Occam Algorithms," Proc. 22nd ACM Symp. on Theory of Computing, pp. 929-965, September 1989. Also available as UIUCDCS-R-89-1544, Dept. of Comp. Sci., Univ. Illinois at Urbana-Champaign, 1989.

Chomsky, Noam, "Formal Properties of Grammars," in Handbook of Mathematical Psychology, ed. R. A. Luce, R. R. Bush and E. Galanter, vol. II, pp. 323-418, Wiley, New York, 1963.

Chomsky, Noam and George A. Miller, "Introduction to the Formal Analysis of Natural Languages," in Handbook of Mathematical Psychology, ed. R. A. Luce, R. R. Bush and E. Galanter, vol. II, pp. 269-321, Wiley, New York, 1963.

Davis, M., Computability and unsolvability, McGraw-Hill, Manchester UK, 1958.

Gold, E. M., "Language Identification in the Limit," Information and Control, vol. 10, pp. 447-474, 1967.

Hamburger, Henry and Ken Wexler, "A Mathematical Theory of Learning Transformational Grammar," J. Mathematical Psychology, vol. 12, pp. 137-177, 1975.

Miller, George A., "Human Memory and the Storage of Information," IRE Trans. on Info. Theory, vol. IT-2, no. 3, pp. 129-137, September 1956.

Miller, George A. and Noam Chomsky, "Finitary Models of Language Users," in Handbook of Mathematical Psychology, ed. R. A. Luce, R. R. Bush and E. Galanter, vol. II, pp. 419-491, Wiley, New York, 1963.

Minsky, M. and S. Papert, Perceptrons, MIT Press, 1969.

Perrault, C. Raymond, "On the Mathematical Properties of Linguistic Theories," Computational Linguistics, vol. 10, no. 3, pp. 165-176, 1984.

Pinker, S., "Formal models of language learning," Cognition, vol. 7, pp. 217-283, 1979.

Pitt, L. and L. G. Valiant, "Computational limitations on learning from examples," J. ACM, vol. 35, pp. 965-984.

Postal, Paul M. and D. Terence Langendoen, "English and the Class of Context-Free Languages," Computational Linguistics, vol. 10, no. 3, pp. 177-181, 1984.

Pullum, G. K. and G. Gazdar, "Natural Languages and Context-Free Languages," Linguistics and Philosophy, vol. 4, pp. 471-504, 1982.

Pullum, G. K., "On Two Recent Attempts to Show that English is Not a CFL," Computational Linguistics, vol. 10, no. 3, pp. 182-185, 1984.

Valiant, L. G., "A Theory of the Learnable," Communications of the ACM, vol. 27, no. 11, pp. 1134-1142, 1984.

Wexler, Kenneth and Peter W. Culicover, Formal Principles of Language Acquisition, MIT Press, Cambridge MA, 1980.


#### 3.1.3 Significance

As indicated above, some of the work here has single-handedly changed the course of history, to negative as well as positive effect.

Minsky (with Papert) showed that there were certain classes of problem which were not learnable with certain networks of perceptrons. Still today (comp.ai newsgroup 19 Oct 90) he is fighting the widespread belief that he killed perceptrons but that now, finally, connectionism has laid to rest "Perceptrons", the book. As he says in comp.ai: "Try reading the book." (which has recently come out in an expanded edition). Actually, he has maintained his research interest in this area over the missing years. Even today there is a need for the theoretical analyses here: the psycholinguistic evidence about how, what and how long we learn (section 3) and the connectionist wave of practical successes (section 4), need to be brought together to reconcile our expectations about what we should be able to build easily with the evidence about what we can actually do easily. Then we will be able to start building systems appropriate to the tasks.

Chomsky and Miller also set Linguistics and Psycholinguistics on a new track with their importation of formal analysis techniques. However, the accuracy of this diagnosis of linguistics does not automatically imply the uniqueness or even soundness of Chomsky's remedy. But the resulting massive persuasion to TGG (section 1.2) has generated a mass of useful research which has given this theoretical approach unprecedented (in Linguistics) opportunity for refinement.

Gold has also set a conundrum for Psycholinguistics. If Natural Language is a Context Free Language (CFL) (see Pullum and Gazdar and Postal and Langendoen) and if our parents don't provide us with the criticism necessary to learn a CFL and our environment doesn't somehow provide us input in a textbook order (see section 3 - Psycholinguists are convinced these conditions aren't satisfied) then we cannot learn Natural Languages.

Chomsky's answer was that we don't learn language, but select a subset of an innate super-language to be appropriate to our language environment. The refinement of Chomsky's TGG approach has lead to the proposal of parameterizable innate rules which exclude a mass of possibilities whilst allowing a measure of variability. This opens the door to a whole new type of learning, and raises a whole lot more questions about the physical mechanism involved.

Another approach is to see if we can come up with a closer classification of Natural Language and the Psycholinguistic and Environmental restrictions.

### 3.2 Development of effective classifications of language.

#### 3.2.1 Introduction

Part of the problem with formal theory is the lack of evidence that the theoretical classification of language relates to the actual human languages and cognitive restrictions. Some basic assumptions are clearly suspect or at least oversimplifications. Do we need to develop new ways of formally characterizing language in terms of the restrictions and heuristics which shape human learning of language?

The lack of references under this head is indicative of a significant lacuna.

#### 3.2.2 Bibliography


Yngve, Victor H., "The Depth Hypothesis," Proc. Symposia in App.

Math., vol. XII, pp. 130-138, Amer. Math. Soc., 1961.

### 3.2.3 Significance

Miller, cited earlier for his work with Chomsky, made another important contribution: on the Magic Number Seven. This is included in the next section. Yngve was the first to put such restrictions to positive effect, implicitly restricting the class of languages he was trying to analyze.

### 3.3 Goals and Issues

GOAL: Theoretical analysis is need to determine and characterize the relation between supervision level, computational constraints, formal language class and base level knowledge.

ISSUE: The positive effect of negative constraints on the computational capacity has been neglected. Such constraints effectively define new subclasses of languages learnable by a given algorithm. The languages humans encounter are not arbitrary but are shaped by our algorithms, limitations and environmental (including supervisory) conditions, being limited to what can be learned (or, stronger still, invented) under these conditions.

### 3.3.1 In this volume

In this volume, Janet Fodor looks at modifications to Phrase Structure Grammar which promise to make it learnable. Sanjay Jain and Arun Sharma look at restrictions which define a reasonable and learnable language class, whilst Leona Fass provides a new representation for Context Free Languages which defines a minimal model which is indeed learnable using inductive methods.

## 4. COGNITIVE SCIENCE

### 4.1 Psychological results on language and restrictions.

Psychological results on language and restrictions are seen as a major foundation for MLNL, with the hope that old and new results and critiques from Psycholinguistics will inspire those who are looking for solutions to problems, and ideas they can implement. For participation in the symposium, it is not necessary that the participant has himself worked on learning programs, but the relevance of his work to such efforts should be made clear.

### 4.2 Linguistic results on the nature of natural language.

Similar considerations apply here. Comparative advice about linguistic theories or formalisms, with critical evaluation on the basis of computability, are basic to MLNL research. Implementers who have adopted a particular linguistic heritage are particularly asked to comment on the reasons for the choice plus the appropriateness in retrospect.

### 4.3 Bibliography

*The emergence of Cognitive Science in the 80s as the interdisciplinary counterpart of Artificial Intelligence represents a huge increase in interest in the potential interdisciplinary contributions to understanding and modeling intelligence, learning and language. This is reflected here in only token form, allowing the reference to the older expositions which preempted the universalist approach and the debates which ensued and lead directly to the recognition of Cognitive Science. The linguistic and philosophical traditions have been to a greater or lesser extent reflected in the last section; whilst the new age neural developments are reflected in the next section to the extent that they are treated at all. This leaves, in the main, Psycholinguistics.*


Anderson, John R. and G. H. Bower, Human Associative Memory, Winston, Washington, 1973.

Anderson, John R., Language, Memory, and Thought, Lawrence Erlbaum Associates, Hillsdale NJ, 1976. Anderson, John R., The Architecture of Cognition, Harvard University, Cambridge MA, 1983.

Anzai, Y. and H. A. Simon, "The theory of learning by doing," Psychology Review, vol. 86, pp. 124-140.

Beckwith, R., C. Fellbaum, D. Gross and G. A. Miller, "WordNet: A Lexical Database Organized on Psycholinguistic Principles", to appear in U. Zernik, Using Online Resources to Build a Lexicon, Erlbaum: NJ, 1990.

Bickerton, Derek, "Creole Languages," Scientific American, vol. 219, no. 1, pp. 108-115, July 1983.

Brown, Roger and Ursula Bellugi, "Three Processes in the Child's Acquisition of Syntax," in Contemporary Issues in Developmental Psychology, ed. E. Endler, L. Boulter, and H. Osser, pp. 411-425, Holt, Rhinehart and Winston, New York, 1968. Reprinted from Harvard Educational Review, 1964, Vol 34, pp 133-151.

Brown, Roger, Psycholinguistics, Free Press, New York NY, 1970. Including: The Child's Grammar from I to III, The First Sentences of Child and Chimpanzee, The 'Tip of the Tongue' Phenomenon.

Brown, Roger, A First Language: the early stages, Allen and Unwin, London UK, 1973.

Cofer, C. N. and B. S. Musgrave, Verbal Behavior and Learning: Problems and Processes, McGraw-Hill, New York NY, 1963.

Derrick, J., The Child's Acquisition of Language, National Foundation for Education Research, Windsor, Berkshire, UK, 1977.

Fletcher, P. and M. Garman, Language Acquisition: Studies in First Language Development, Cambridge University Press, Cambridge UK, 1979.

Fodor, Janet Dean, "Constraints on gaps: is the parser a significant influence?," Linguistics, vol. 21, no. 1, pp. 9-35, 1984.

Fodor, Jerry A., The Language of Thought, MIT Press?, 1975.

Fraser, D., U. Bellugi, and R. Brown, "Control of Grammar in Imitation, Comprehension, and Production," Journal of Verbal Learning and Verbal Behaviour, vol. 2, pp. 121-135, 1963.

Gentner, D., "Some interesting differences between nouns and verbs," Cognition and Brain Theory, vol. 4, pp. 155-184, 1982.

Howell, Peter and Stuart Rosen, "Natural auditory sensitivities as universal determiners of phonemic contrasts," Linguistics, vol. 21, no. 1, pp. 205-235, 1984.

Hubel, D. H., "The Brain (Introduction to Special Issue)," Sci. Amer., vol. 241, no. 3, pp. 38-47, September 1979.

Hubel, D. H. and T. N. Wiesel, "Brain Mechanisms of Vision," Sci. Amer., vol. 241, no. 3, pp. 130-145, September 1979.

Huey, E. B., The Psychology and Pedagogy of Reading, MIT Press, Cambridge MA, 1908/1968.

Huttenlocher, Janellen, Patricia Smiley, and Rosalind Charney, "Emergence of Action Categories in the Child: Evidence from Verb Meanings," Psychological Review, vol. 90, no. 1, pp. 72-93, 1983.

Jakobovits, L. A., Foreign Language Learning, Newbury House, Rowley, Massachusetts, 1970.

Klopf, A. Harry, The Hedonistic Neuron: A Theory of Memory, Learning and Intelligence, Hemisphere, WASHINGTON DC, 1982.

Kuczaj, Stan A., Crib Speech and Language Play, Springer-Verlag, New York NY, 1983.

Lakoff, George and Mark Johnson, Metaphors we Live By, University of Chicago Press, 1980.

Lenneberg, Eric H., Biological Foundations of Language, Wiley, New York, 1967.

Lindblom, Bjorn, Peter MacNeilage, and Michael Studdert-Kennedy, "Self-organizing processes and the explanation of phonological universals," Linguistics, vol. 21, no. 1, pp. 181-203, 1984.

MacWhinney, B., Mechanisms of Language Acquisition, Lawrence Erlbaum Associates, Hillsdale NJ, 1986.

Mehler, J., P. Jusczyk, G. Lambertz, N. Halsted, J. Bertoncini, and C. Amiel-Tison, "A precursor of language acquisition in young infants," Cognition, vol. 29, pp. 143-176, 1988.

Miller, George A., "The magical number seven, plus or minus two: Some limits on our capacity for processing information," Psych. Rev., vol. 63, pp. 81-97, 1956. Republished in George A. Miller, 'The Psychology of Communication.', 1967.

Miller, George A. and Susan M. Ervin, "The Development of Grammar in Child Language," in Acquisition of Language, ed. U. Bellugi and R. Brown, Society for Research in Child Development Monographs, 1965.

Miller, George A., The Psychology of Communication, Allen Lane: Penguin Press, London, 1967.

Moore, Timothy E., Cognitive Development and the Acquisition of Language, Academic Press, New York NY, 1973.

Nelson, K., "Concept, Word and Sentence," Psych. Rev., vol. 8, pp. 267-285, 1974.

Newell, Allen and Herbert A. Simon, "GPS: A program that simulates human thought," in Computers and Thought, ed. E. Feigenbaum and J. Feldman, pp. 279-293, McGraw-Hill, 1963.

Newell, A. and H. Simon, Human Problem Solving, Prentice-Hall,

Englewood Cliffs NJ, 1972.
Oller, D. Kimbrough, "Simplification as the Goal of Phonological Processes in Child Speech," Language Learning, vol. 24, no. 2, pp. 299-303, 1974.
Oller, Jr, John W., "Language as Intelligence," Language Learning, vol. 31, no. 2, pp. 465-492, 1981.
Piaget, Jean, The Language and Thought of the Child, Humanities Press, New York, 1926/1959. Companion to and precursor of 'Judgement and Reasoning in the Child', 1927.
Piatelli-Palmarini, M., Language and Learning: The Debate between Jean Piaget and Noam Chomsky, Routledge and Kegan Paul, London, England, 1979.
Piatelli-Palmarini, M., "Evolution, selection and cognition: From biology and in the study of language," Cognition, vol. 31, pp. 1-44, 1989.
Pike, Kenneth L., Language in Relation to a Unified Theory of the Structure of Human Behavior, Mouton, The Hague, Holland, 1954/1967.
Pinker, Steven, "Concept, Word, and Sentence: Interrelations in Acquisition and Development," Cognition, vol. 7, pp. 217-183, 1979.
Pylyshyn, Z., "The role of location indexes in spatial perception: A sketch of the FINST spatial-index model," Cognition, vol. 32, pp. 65-97, 1989.
Riegel, K. F., "The Language Acquisition Process: A Reinterpretation of Selected Research Findings," in Life-Span Developmental Psychology: Research and Theory, ed. P. B. Baltes, pp. 357-399, 1970.
Slobin, Dan I., "Imitation and Grammatical Development in Children," in Contemporary Issues in Developmental Psychology, ed. E. Endler, L. Boulter, and H. Osser, pp. 437-443, Holt, Rhinehart and Winston, New York, 1968.
Slobin, Dan I., The Ontogenesis of Language, Academic Press, New York, 1971.
Sloman, Aaron and Monica Croucher, "Why Robots will have Emotions," 7th International Joint Conference on Artificial Intelligence, pp. 197-202, 1981.
Smith, F. L. and George A. Miller, The Genesis of Language: A Psycholinguistic Approach, MIT Press, Cambridge, Massachusetts, 1966.
Suppes, Patrick, "The Semantics of Children's Language," American Psychologist, pp. 103-114, 1974.
Turk, C. C. R., "A Correction NL Mechanism", ECAI'84: Advances in Artificial Intelligence, ed. T.O'Shea, Elsevier: Amsterdam, 1984.
Vetter, H. J. and R. W. Howell, "Theories of Language Acquisition," Jnl of Psycholinguistic Res., vol. 1, no. 1, pp. 31-64, 1971.
Viberg, Ake, "The verbs of perception: a typological study," Linguistics, vol. 21, no. 1, pp. 123-162, 1984.
Widerstrom, Anne, "Mothers' Language and Infant SensoriMotor Development: Is there a Relationship," Language Learning, vol. 32, no. 1, pp. 145-166, 1982.

### 4.4 Significance

Having worked so hard to promote interdisciplinary connections and establish Cognitive Science, this is not the place to try to disentangle the woven threads into the distinct fields. The work and the researchers presented here is increasingly moving beyond the primary boundaries of the host discipline.

By far the majority of the references included here can, however, be considered as having a psycholinguistic orientation, a bias which is by no means independent of our focus on MLNL. And Piaget can be considered the father of Psycholinguistics. The reference here is actually the first of a dozen or more books on language and reasoning 'chez l'enfant'.

It goes without saying that the work of Anderson, Brown, and Newell and Simon is essential reading: Anderson for work on Memory, which has extended in to Language Acquisition models (section 7); Brown for the enormous contribution he and his coworkers have made to the analysis of child language and the direction of the field; and Newell and Simon for their work on problem solving which was actually fundamental to the inception of AI - it is often forgotten that the cognitive science elements were there from the beginning.

Pike's broad view of language is a mammoth effort showing incomparable insight into linguistic processes, and like Piaget worth making the effort to digest. The work of Lakoff and Johnson on metaphor is classic and is also essential reading - metaphor and metonymy are not just linguistic devices but are fundamental to the way we us language and understand the world. Language as we know it, ontology, just couldn't exist without extending our experience of particular situations to others. Metaphor and paradigm give us a model for employing contrast and similarity as a basis for generalization and application of knowledge. The work of Clark and Clark examines particular aspects of fundamental metaphorical usage in language and is again essential reading.

Along with a myriad of other fundamental work, the Clarks' papers are to be found in some key compendia. Six such volumes are included in the bibliography above. We select out some of the work of particular interest from them here:

Cofer and Musgrave (1963) includes Roger Brown and Colin Fraser on The Acquisition of Syntax.

Smith and Miller (1966) includes Jerry A. Fodor (How to Learn to Talk: Some Simple Ways), Eric H. Lenneberg (The Natural History of Language), David McNeill (Developmental Psycholinguistics) and Dan I. Slobin (Acquisition of Russian as a Native Language)

Slobin (1971) includes Martin D. S. Braine (On Two Types of Models of the Internalization of Grammars), David McNeill (The Capacity for the Ontogenesis of Grammar) and David S. Palermo (On Learning to Talk: Are Principles Derived from the Learning Laboratory Applicable?).

Moore (1973) includes Melissa Bowerman (Structural Relationships in Children's Utterances: Syntax or Semantic?), Eve V. Clark (What's in a Word? On the Child's Acquisition of Semantics in his First Language), Herbert H. Clark (Space, Time, Semantics, and the Child), Susan Ervin-Tripp (Some Strategies for the First Years) and Gary M. Olson (Developmental Changes in Memory and the Acquisition of Language) and H. Sinclair-deZwart (Language Acquisition and Cognitive Development).

Fletcher and Garman (1979) includes contributions by Melissa Bowerman (The Acquisition of Complex Sentences), Bruce L. Derwing (Language Acquisition: Studies in First Language Development), Eve V. Clark, (Building a Vocabulary: Words for Objects, Actions and Relations), William J. Baker, (Recent Research on the Acquisition of English Morphology), Robert Grieve and Robert Hoogenraad (First Words), Patrick Griffiths (Speech Acts and Early Sentences) and Michael P. Maratsos (Learning How and When to Use Pronouns and Determiners).

MacWhinney (1986) includes E. Clark (The principle of contrast: A constraint on language acquisition), P. Langley & J. Carbonell (Language Acquisition and Machine Learning), B. MacWhinney & J. Sokolov (Acquiring syntax lexically) and S. Pinker (The bootstrapping problem in language acquisition).

In these cases there would be too much to address each contribution, but the volumes are thoroughly worth a browse through, looking particularly at the papers mentioned.

As a final topic in Psycholinguistics, we mention the particular focus of imitation and correction, including the role of language play. Fraser, Bellugi and Brown have performed fundamental studies here which have some surprising results about the relative difficulty of production, comprehension and imitation. Kuczaj and Derrick also extend the ideas of imitation and reduction to consider the child's own spontaneous paradigmatic production and self-imitation, and the parents use of imitation and expansion. Again there may be some surprises about just what parental language is most, and least, helpful.

There is one other collection of papers, Piatelli (1979), which is a must, and arises from consideration of the imitation and correction paradigm and its insufficiency. Here the protagonists of the great debate of nativism versus constructivism address each other's position directly in position papers and in reply. These deals directly with the issues that came out of the theoretical considerations of section 3 and represents the point of time, around the birth point of Cognitive Science, where the possibilities for reconciling theoretical and empirical results on language learning were

just beginning to re-emerge in the face of rampant nativism.

Moving from those papers directly concerned with language learning, we come first to the generalization where a whole culture and language is incompletely learned and generates pidgins and creoles. In such a context where the learners come from a mix of language backgrounds and learn the words but not the grammar of a new "common" language, a new creole grammar emerges which is relatively independent of all of the original languages. Bickerton presents an interesting paper on this phenomenon. It would seem that it should contribute to our modelling of default preferences during language learning, and that it should be contrasted with child grammars. These challenges have yet to be taken up.

Moving further afield, we come to the famous paper of Miller on the "Magic Number Seven", which challenges us to take our known Cognitive Restrictions into account, and is really the key ingredient in finding a solution to the innateness debate and the theoretical conundra of section 3. And then there is Huey's classic on reading - the only work from last century cited in this review. Techniques of following eye motion and examining our reading behaviour can also provide insights into language behaviour, and help explain some of the behaviour of our learning programs too.

Extending from MLNL to MLNLO, leads us to consider important work related to our ontology, and of course the visual modality which we feel is so dominant and which is one of the most well explored areas of cognition. Here the work of Hubel and Wiesel is again classic - and has been the basis for some experiments on self-organization neural models, both for vision and language (Powers, section 4), whilst the work of Pylyshyn is directly complementary to some Psycholinguistic studies.

### 4.5 Goals and Issues

GOALS: To provide the empirical evidence for the roles of innate knowledge and specific and general learning mechanisms, as well as for environmental conditions including parents and other human supervisors and critics plus the physical laws and feedback deriving from physiological constraints.

ISSUE: How much is (necessarily) innate? From how minimal a base state can learning be effective in bootstrapping?

ISSUE: How much supervision, teaching and criticism is necessary for effective learning? To what extent can a reactive environment substitute? What cognitive constraints shape our languages?

#### 4.5.1 In this volume

In this volume, Mallory Selfridge addresses the question of how children learn to recognize ungrammatical sentences - the question of negative information again. James Martin looks at the problem of how children acquire and distinguish the manifold metaphors which are part and parcel of language. By way of contrast, Steven Lytinen and Carol Moon consider second language acquisition, bootstrapping from one language to another.

## 5. PARALLEL NETWORKS

### 5.1 Neural models of parsing and learning.

There is a separate parallel symposium on "Connectionist Natural Language Processing". For reason we are most concerned here with the advantages of neural approaches over conventional machine learning OR with deep modelling of neurolinguistic processes, rather than with application of backpropogation in this or that area - there is really just too much of an explosion in Connectionism to do justice to it here - we provide the fundamental references but no more. But we explore in other directions. In particular, we are interested in hard neurological evidence and the associated theories.

### 5.2 Parallel models of parsing and learning.

Implementations on parallel hardware are also of interest, as are parallel or parallelized algorithms and theoretical contributions on the role, parallelism, backtracking etc. in language and learning processes.

The interest in parallel parsing goes back just as far as the roots of connectionism - in fact there has been a long standing assumption that natural language parsing was inherently parallel. This debate is starting to favour the view that it is not, even with out backtracking. But there is still evidence that our own brains do use at least a partly parallel process.

### 5.3 Bibliography

*This very short list points to both the old school and the new age of associative and neural networks, as well as the only parallel language learning proposals I am aware of.*


Amari, S. and M. A. Arbib, Competition and Cooperation in Neural Nets, Springer-Verlag, Berlin GDR, 1982.

Charniak, E. and E. Santos, "A connectionist context-free parser which is not context-free, but then it is not really connectionist either," Proc. 9th Conf. of the Cog. Sci. Soc., pp. 70-77, Seattle WA, July 1987.

Fodor, J. A. and Z. W. Pylyshyn, "Connectionism and cognitive architecture: A critical analysis," Cognition, vol. 3, pp. 3-72, 1988.

Gigley, H. M., "Artificial Intelligence meets Brain Theory: An Integrated Approach to Simulation Modelling of Natural Language Processing," Proceedings of the Sixth European Meeting on Cybernetics and Systems Research, North-Holland, 1982.

Gigley, H. M., Neurolinguistically Constrained Simulation of Sentence Comprehension: Integrating Artificial Intelligence and Brain Theory, Ph.D. Thesis, University of Massachusetts, Amherst Massachusetts, 1982.

Gigley, H. M., "From HOPE en l'ESPERANCE: On the Role of Computational Neurolinguistics in Cross-Language Studies," Proceedings of Coling84, pp. 452-456, Association for Computational Linguistics, 2-6 July 1984.

Grossberg, S., "Contour Enhancement, Short Term Memory, and Constancies in Reverberating Neural Networks," Stud. App. Math., vol. LII, no. 3, pp. 213-257, 1973.

Grossberg, S., "Adaptive Pattern Classification and Universal Recoding: I. Parallel Development and Coding of Neural Features," Biol. Cyb., vol. 23, pp. 121-134, 1976. Sequel: II. Feedback, Expectation, Olfaction, Illusions (pp. 187-202)

Grossberg, S., "On the Development of Feature Detectors in the Visual Cortex with Applications to Learning and Reaction-Diffusion Systems," Biol. Cyb., vol. 21, pp. 145-159, 1976.

Hebb, D. O., Organization and Behaviour, Wiley, New York, 1949.

Hinton, G. E., "Representing part-whole hierarchies in connectionist networks," Proc. 10th Conf. of the Cog. Sci. Soc., pp. 48-54, Montreal, 1988.

Holbach-Weber, Susan, "Connectionist Models and Figurative Speech", DFKI TM-89-01, Deutsches Forschungzentrum fuer KI, Saarbruecken FRG 1989.

Jain, Sanjay and Arun Sharma, "Language Learning by a 'Team'," Proc. ICALP'90, 1990.

Kohonen, T., "A Simple Paradigm for the Self-Organized Formation of Structured Feature Maps," in Competition and Cooperation in Neural Nets, ed. S. Amari and M. A. Arbib, pp. 248-266, Springer-Verlag, 1982.

Kohonen, T., "Self-Organized Formation of Topologically Correct Feature Maps," Biol. Cyb., vol. 43, pp. 59-69, 1982.

Kohonen, T., "Analysis of a Simple Self-Organizing Process," Biol. Cyb., vol. 44, pp. 135-140, 1982.

Lachter, J and T. G. Bever, "The relation between linguistic structure and associative theories of language learning - A constructive critique of some connectionist learning models," Cognition, vol. 28, pp. 195-247, 1988.

Longuet-Higgins, H. C., David J. Willshaw, and O. P. Buneman, "Theories of Associative Recall," Qtly Revs Biophysics, vol. 3, no. 2, pp. 223-244, 1970.

Malsburg, C. von der, "Self-Organization of Orientation Selective Cells in the Striate Cortex", Kybernetik, vol. 14, pp. 85-100, 1973.

Pinker, Steven and A. Prince, "On language and connectionism: Analysis of a parallel distributed processing model of language acquisition," Cognition, vol. 3, pp. 73-193, 1988.

Pollack, J. B., "Cascade back-propagation on dynamic connectionist networks," Proc. 9th Mtg of Cog. Sci. Soc., pp. 391-404, Seattle WA, 1987.

Pollack, J. B., "Connectionism: past, present and future," AI Review, vol. 3, pp. 3-20, 1989.

Powers, David M. W., "Neurolinguistics and Psycholinguistics as a

Basis for Computer Acquisition of Natural Language," SIGART, no. 84, pp. 29-34, June 1983. Also DCS Report 8301, Dept of Computer Science, University of NSW, Australia (Abstract: Aust. Postgrad. Research Conf., Feb. 1983).
Rumelhart, D. E. and J. L. McClelland, Parallel Distributed Processing: Experiments in the Microstructure of Cognition, MIT Press, Cambridge MA, 1986.
Selman, B., "Connectionist systems for natural language understanding," AI Review, vol. 3, pp. 23-31, 1989.
Sharkey, N. E., "Fast connectionist learning: words and case," AI Review, vol. 3, pp. 33-47, 1989.
Sutton, R. S. and A. G. Barto, "Towards a Modern Theory of Adaptive Networks: Expectation and Prediction," Psych. Rev., vol. 88, no. 2, pp. 135-170, 1981.
Uhr, Leonard, Pattern Recognition, Learning, and Thought: Computer-Programmed Models of Higher Mental Processes, Prentice-Hall, Englewood Cliffs NJ, 1973.
Waltz, D. L. and J. B. Pollack, "Massively parallel parsing," Cognitive Science, vol. 9, pp. 51-74, 1985.

### 5.4 Significance

To start with modern Connectionism, Rumelhart and McClelland and their PDP group have put out three volumes, the last with a disk of sample programs. For a general review of the field see Pollack (1989), and in relation to natural language applications, see Selman (1989) - the other articles in the same issue of AI Review as these may also be of interest.

To go back to the roots, Hebb originated the plasticity hypotheses which is largely the basis for the PDP work, with the addition of backpropogation. Without this feedback, useful learning can still be done as shown by von der Malsburg in reproducing the visual cortical columns. Powers has applied the same technique to language between the grapheme and noun phrase levels.

Grossberg has been pursuing the properties of recurrence for well over a decade, originally as a model of memory, but more recently has come up with some interesting results in feature recognition. His recent book is not listed but presents his whole program through the papers he has presented over the years. Kohonen and Longuet-Higgins et al. also represent older schools concerned with associative memory properties. Amari and Arbib provides a good time-stamp for the point just as Connectionism began to take off and emphasises the competitive and cooperative aspects which have been taking a back seat recently, but produce useful results when applied to appropriate problems.

Uhr is pretty good for its time, but rather dated now, and somewhat negative on the Perceptron/Self-Organization question. Waltz and Pollack is very aware of the neural aspects, whilst the parallelism of Jain and Sharma is totally independent of connectionism - their team model being more related to the OR-parallelism of logic programming or evolutionary learning.

### 5.5 Goals and Issues

GOAL: Neural investigations need to determine and characterize the nature and role of the human (animal) wetware, as well as stretching the limits of neural inspired models.
ISSUE: What are the limits of genetic determination, boundary conditions and self-organizational determination?
ISSUE: Neural simulations to date tend to be passive recognizers reacting to the sensory-motor input. Does there exist some sort of active learning which is different, which is not just a feedback control system, but capable of initiating behaviour?
ISSUE: How does all of this relate to language? Is language just a consequence of our neural capacities in combination? Or are language specific neural level mechanisms to be found?

#### 5.5.1 In this volume

In this volume we present some papers which address the appropriateness of connectionist methods for MLNLO, and which were presented in a joint session with the AAAI Spring Symposium on Connectionist Natural Language Processing. These include papers by Jane Hill trying to get the best of both worlds with hybrid models, Andreas Stolcke looking at the relative merits of the different approaches and the blurring of the boundaries, and David Powers suggests that some connectionist and AI learning techniques are overkill if applied indiscriminately at all levels.

Jan Scholtes and David Powers present work using self-organizational rather than PDP-style models, observing how rules and classes can be related to synapses and neurons, and how basic linguistic properties emerge automatically using these simple nets.

Stevan Harnad takes up the story of what might happen with these emergent categories, and how they could form the bridge into a symbolic learning system. Bartell and Cottrell also consider the relationship between Connectionism and Symbol Grounding, exploring it with a simulated billiard table. Honavar actually uses multiple connectionist networks to provide an interacting language community.

## 6. SYMBOL GROUNDING

### 6.1 Grounding of Natural Language Systems.

Where is the border between syntax and semantics? When can a system be said to know something as opposed to just churning out a pat response? Does learning provide an answer to these old chestnuts? How far can you get with an ungrounded system?

### 6.2 Interaction between Modalities and Learning of Ontology.

We are particularly interested in contributions in which aspects of language are learnt and used in a context or where language input and output are supplemented by (or indeed supplement) other forms of interaction between the language system and the environment in which it is embedded. The system could be a robot, simulated or actual; the environment could be provided by a vision system; or we could have a humbler interface to a database, an operating system or other application.

### 6.3 Bibliography

*The Cognitive Science and Theoretical Approaches literature is relevant background, the work listed here faces directly the question of the individual in relation to his world and his representation thereof. Explicit reference to Searle and Turing are avoided here as irrelevant, but you can't explore far before falling over the ubiquitous Chinese Room.*

Block, H. D., J. Moulton, and G. M. Robinson, "Natural Language Acquisition by a Robot," Int. J. Man-Mach, Stud., vol. 7, pp. 571-608, 1975.
Brewster, E. Thomas and Elizabeth S. Brewster, Language Acquisition Made Practical: Field Methods for Language Learners, Lingua House, Colorado Springs, 1976.
Carbonell, J. G. and G. Hood, "The World Modelers Project: Objectives and simulator architecture," in Machine Learning: a Guide to Current Research, Kluwer, Boston MA, 1986.
Dennet, D. C., "Intentional systems in cognitive ethology," Behavioral and Brain Sciences, vol. 6, pp. 343-90, 1983.
Fodor, Jerry, Psychosemantics, MIT Press, Cambridge MA, 1987.
Gibson, J. J., An ecological approach to visual perception, Houghton Mifflin, Boston MA, 1979.
Halliday, M. A. K., Learning How to Mean, Edward Arnold, London UK, 1975.
Harnad, Stevan, "Metaphor and mental duality," in Language, mind and brain, ed. T. Simon and R.
Scholes, Lawrence Erlbaum, Hillsdale NJ, 1982. Harnad, Stevan, Categorical perception: The groundwork of Cognition, Cambridge University Press, New York NY, 1987.
Hayes, P. J., "The Naive Physics Manifesto," in Expert Systems in the Micro-electronics Age, ed. D. Michie, pp. 242-270, Edinburgh U.P., Edinburgh, Scotland., 1979.
Hume, David, Creating Interactive Worlds with Multiple Actors, B.Sc. Honours Thesis, Electrical Engineering and Computer Science, University of New South Wales, Sydney, AUSTRALIA, November 1984. (Supervisors: David M. W. Powers, Graham B. McMahon.).
Jolley, J. L., The Fabric of Knowledge: a study of the relations between ideas, Duckworth, London UK, 1973. (Interesting but requires many grains of salt.)
Lakoff, George and Mark Johnson, Metaphors we Live By, University of Chicago Press, 1980.
Lenat, D. and R. V. Guha, "The world according to CYC," ACA-AI-300-88, MCC, 3500 West Balcones Center Drive, Austin TX. To be published by Addison-Wesley in expanded form as "Building large knowledge based systems".
Lenneberg, Eric H., "Understanding Language without Ability to

Speak: A Case Report," in Contemporary Issues in Developmental Psychology, ed. E. Endler, L. Boulter, and H. Osser, pp. 403-411, Holt, Rhinehart and Winston, New York, 1968. Reprinted from Journal of Abnormal and Social Psychology, 1962, Vol 65, pp 419-425.

Lerner, E. J., "Computers That See," IEEE Computer, vol. 17, no. 10, pp. 28-33, October 1980.

Lettvin, J. Y., H. R. Maturana, W. S. McCulloch, and W. H. Pitts, "What the Frog's Eye Tells the Frog's Brain," Proceedings of the Institute of Radio Engineers, vol. 47, no. 11, pp. 1940-1951, November 1959.

Marshall, John C., "Language Acquisition in a Biological Frame of Reference," in Language Acquisition: Studies in First Language Development, ed. P. Fletcher and M. Garman, pp. 437-453, Cambridge University Press, Cambridge UK, 1979.

McCarthy, J., L. D. Earnest, D. R. Reddy, and P. J. Vicens, "A Computer with Hands, Eyes, and Ears," AFIPS Conf. Proc. Fall JCC 1968, vol. 33:1, pp. 329-338, 1968.

Newell, A., "Physical Symbol Systems," Cognitive Science, vol. 4, pp. 135-83, 1980.

Piaget, Jean, The Child's Conception of the World, Kegan Paul, Trench, Truber and Co., London UK, 1929.

Piaget, Jean, The Construction of Reality in the Child, Basic Books, New York, 1954. Original Title: 'La Construction du Reel chez l'Enfant.

Powers, David M. W., "Robot Intelligence," Electronics Today International (Australia), pp. 15-18, December 1983.

Pribram, K. H., Languages of the Brain, Prentice-Hall, Englewood Cliffs, NJ, 1971.

Pustejovsky, James, "On the acquisition of lexical entries: The perceptual origin of thematic relations," Proc. 25th Ann. Mtg of the Association for Computational Linguistics, pp. 172-178, 1987.

Pylyshyn, Z. W., The robot's dilemma: The frame problem in artificial intelligence, Ablex, Norwood NJ, 1987.

Reeker, Larry, "The interplay of semantic and surface structure acquisition," in Recent Advances in the Psychology of Language, ed. R. Campbell and P. Smith, vol. 2, pp. 71-90, Plenum Press, 1978.

Sammut, Claude and David Hume, "Learning concepts in a complex robot world," in Machine Learning: a Guide to Current Research, Kluwer, Boston MA, 1986. Shepard, R. N. and L. A. Cooper, Mental images and their transformations, MIT Press, Cambridge MA, 1982.

Sloman, Aaron and Monica Croucher, "Why Robots will have Emotions," 7th International Joint Conference on Artificial Intelligence, pp. 197-202, 1981. Tanz, Christine, Studies in the Acquisition of Deictic Terms, Cambridge University Press, Cambridge UK, 1980.

Turbayne, C. M., The Myth of Metaphor, University of South Carolina Press, Columbia, South Carolina, 1971. 2nd Ed.

Wales, Roger, "Deixis," in Language Acquisition: Studies in First Language Development, ed. P. Fletcher and M. Garman, pp. 241-260, Cambridge University Press, Cambridge UK, 1979.

Widerstrom, Anne, "Mothers' Language and Infant SensoriMotor Development: Is there a Relationship," Language Learning, vol. 32, no. 1, pp. 145-166, 1982.

### 6.4 Significance

The catch-phrase "Symbol-Grounding" has been popularized primarily by Harnad, who has been interested in it from a philosophical view and as an exponent of Total Turing Test versus the Turing Test (in relation to Searle's Chinese Room). The problem is that no matter how many time you translate your text to a new "representation language" you still have a language composed of symbols and no possibility of real meaning or understanding. Where does our meaning and understanding come from?

The work referenced here is quite varied, varying from a text on how to learn a language monolingually (Brewester and Brewester), to various world modelling projects, including Lenat's CYC project and Carbonell and Hood's World Modelers Project. Lenat sees the problem of representation as being much more pressing than learning at this time, and his project is not supposed to start its automatic acquisition phase till 1994.

On a smaller scale, Hume and Powers have developed a Robot World modelling system for language learning work (used also by Sammut for concept learning; see also section 7), and many others have worked with or proposed similar schemes, including, as early examples, Block et al. and McCarthy et al. The Naive Physics Manifesto of Hayes encourages moving to the big time - toy world's are only good for toy systems, so sometime we have to represent the world more realistically.

Others researchers have concentrated on particular problems or manifestations - this includes the work of Sloman and Croucher and that of Pylyshyn.

Moving further back to the fundamental psycholinguistic and neurolinguistic study of how we develop our ontology and semantics, we come to some classics. Lettvin et al. on frog vision, and Piaget and Lenneberg on language understanding in normal and abnormal circumstances respectively. Lerner and Marshall provide more recent perspectives. Then there is Jolley's whimsical attempt to catalogue the whole of knowledge - on the basis of similarities which run orthogonally across all areas and levels of knowledge.

Again metaphor is an important part of the answer to symbol grounding. Once the world has made an iconic image somewhere in our brain, we have an imperfect reflection of reality. We continue to abstract and manipulate this in a way determined by our experience, that is by similarity to what we have experienced in the past, in the same or in different modalities, and always in at least slightly different contexts. The frame problem arises when we make our concepts to small and forget that in fact we tend to be present whole frames in each modality, and it is these we compare and process.

Irrespective of whether Harnad is right about characterizing this as being the major problem for Natural Language and Artificial Intelligence, Symbol Grounding is currently one of the weakest areas, and these few pointers here need to be paid more attention and to grow into resources for future MLNLO work.

### 6.5 Goals and Issues

ULTIMATE GOAL: To have language used effectively by the computer for the purpose we intend.

ISSUE: When are we just translating from one language to another? When are we doing more: understanding, communicating, intending? Where does a computer derive its motivation from? Its programmer? Where do we derive our motivation from?

TOY SUB GOAL: To provide a toy environment in which the above is achieved.

REAL SUB GOAL: To achieve this in an actual application environment.

ISSUE: How similar a sensory-motor environment and perceptual interface to ours is needed to allow learning of language? And what criterion do we learn to?

#### 6.5.1 In this volume

Harnad presents the Symbol Grounding Problem, and consider Neurological and Neural Network findings which support a thesis these networks have natural classification properties, such that the categories which arise could form the ground level for a symbol system.

Siskind argues also for a solid lexical semantics in the form of a naive physics and introduces a system providing such a mechanism. Weber and Bartell also tackles this problem: Weber in the context of toy domain involving geometric objects; Bartell in the dynamic domain provided by a billiard table simulation. Jeffrey Siskind acquires new word meanings from dynamic conjunction of sequences of conceptual structures and correlated language input.

Honavar looks at learning across multiple modalities in a dynamic community of simulated language users, being born, living a while and dying. He argues for the need for learning, vision and language to be treated together.

It is also possible to learn new semantics second hand, from a dictionary, or from usage in context. Brent and Zernik looks at what can be acquired from a corpus, examining the range of usage of a word. Hearst seeks to make use both of a machine readable dictionary and corpus data.

Peter Hastings and Steven Lytinen acquire their semantics in the more conventional context of a IS-A hierarchy, again examining what can be learned by tracking word usage.

## 7. SYSTEM DEVELOPMENT

### 7.1 Computable hypotheses and heuristics for language learning.

"Proposals of how to build a language learning system are few and far between, and will be received with interest, as will more limited argument about the significance of various hypotheses, heuristics or methodologies for language learning implementations."

At the moment language learning work tends to work in a small way, examining how far one can get with certain techniques. Whether these techniques are implemented as AI or Psychological modelling, they should make a contribution to our understanding of language and learning.

### 7.2 Experimental language learning systems and their rationale.

"Reports on successfully implemented language learning systems will be received with amazement! Characterizations of what can be learnt by the system, or any precursor thereof, should be included, along with explanations of the methodology used."

Even when there are concrete ideas about how to proceed, there has been negligible funding for MLNL research, and the implementations have been limited. But the field does seem to be ripe now for a major effort, and there is no shortage of ideas and methodologies coming from all the disciplinary vantage points considered in this review.

### 7.3 Bibliography

*Currently all the MLNL work I know of lies somewhere between, and less than, providing full proposals or implementations, but research is progressing more or less scientifically, by small steps, as influenced by particular views.*

*I list here every known researcher who has developed any system which in any sense makes a claim to learn an aspect of language. Not every individual publication is listed, but rather the most comprehensive and accessible.*


Anderson, John R., "Computer Simulation of a Language Acquisition System: A First Report," in Information Processing and Cognition: The Loyola Symposium, ed. R. L. Solso, pp. 295-349, Lawrence Erlbaum Associates, Hillsdale, 1975.

Anderson, John R., "A Theory of Language Acquisition Based on General Learning Principles," 7th International Joint Conference on Artificial Intelligence, pp. 97-103, 1981.

Berwick, Robert C., The Acquisition of Syntactic Knowledge, MIT Press, Cambridge MA, 1985.

Berwick, Robert C. and Sam Pilato, "Learning Syntax by Automata Induction," Machine Learning, vol. 2, pp. 9-38, 1987.

Davila, Lazaro, David M. W. Powers, Debbie M. Meagher, and David Menzies, Further Experiments in Computer Learning of Natural Language, pp. 458-468, Sydney NSW Australia, September 1987. Second Australian Conference on Artificial Intelligence, November 2-4, 1987.

Dresher, B. E. and J. D. Kaye, "A computational learning model for metrical phonology," Cognition, vol. 34, pp. 137-195, 1990.

Feigenbaum, E. A., "The simulation of verbal learning behavior," in Computers and Thought, ed. Feigenbaum and Feldman, McGraw-Hill, New York NY, 1963.

Granger, Richard, "FOUL-UP: a program that figures out meanings of words from context," Proc. 5th IJCAI, pp. 172-178, 1977.

Harris, Larry R., "A System for Primitive Natural Language Acquisition," Int. J. Man-Machine Studies, vol. 9, pp. 153-206, 1977.

Kelley, K. L., "Early Syntactic Acquisition," P-3719, Rand Corporation, Santa Monica CA, November 1967.

Kucera, Henry, "The Learning of Grammar," Perspectives in Computing, vol. Vol 1, no. 2, pp. 28-35, 1981.

Lamb, Sydney M., "On the Mechanization of Syntactic Analysis," 1961 Conference on Machine Translation and Applied Language Analysis, vol. II, pp. 674-685, Her Majesty's Stationery Office, London, 1961.

Langley, Pat, "Language Acquisition Through Error Recovery," Cognition and Brain Theory, vol. 5, no. 3, pp. 211-255, 1982.

Lehman, J. F., "Adaptive Parsing: A general method for learning idiosyncratic grammars," Proc. 7th Int'l Machine Learning Conference, Austin TX, 1990.

Lehman, J. F., "Supporting Linguistic Consistency and Idiosyncracy," Proc. 12th Cognitive Science Society Conference, Boston MA, 1990.

McMaster, I., J. R. Sampson, and J. E. King, "Computer Acquisition of Natural Language," Int'l Jnl of Man-Machine Studies, vol. 8, pp. 367-396, 1976.

MacWhinney, B., "Conditions on acquisitional models", Proc. Annual Conference of the ACM, New York, 1978.

Miller, P. L, "An Adaptive Natural Language System that Listens, Asks and Learns," Proc. 4th IJCAI, 1975.

Narasimhan, R., Modelling Language Behaviour, Springer-Verlag, Berlin, 1981.

Powers, David M. W., "Experiments in Computer Learning of Natural Language," Proc. Aust. Comp. Conf., pp. 489-500, Sydney NSW Australia, November 1984.

Powers, David M. W. and Christopher C. R. Turk, Machine Learning of Natural Language, Springer, London/Berlin, December 1989. Based on Powers' Ph.D. thesis, University of NSW, Australia, 1985.

Rayner, Manny, Asa Hugosson, and Goran Hagert, "Using a logic grammar to learn a lexicon," R88001, Swedish Institute of Computer Science, 1988.

Rayner, Manny, "Applying Explanation-Based Learning to Natural Language Processing," R890144 and R890145, 1989. Part 2 with Christer Samuelsson, also 1989.

Reeker, Larry, "The computational study of language acquisition," Advances in Computers, vol. 15, Academic Press, New York NY, 1976.

Reeker, Larry, "Adaptive Individualized User Interfaces for Computerized Processes, Report 1: Project Overview," Report 2 (with L. Warren Morrison): Design Report. Prepared by the BDM Corp., McLean VA, for Applied Information Technology Research Center, Columbus OH, 1988.

Rolandi, Walter G., "Language Acquisition by Machine: An Operant Investigation," Masters Thesis, Dept. of Psychology, Univ. Sth Carolina, 1988. (Supervisor: James B. Appel)

Salveter, Sharon C., "Inferring conceptual graphs," Cognitive Science, vol. 3, no. 2, pp. 141-166, 1979.

Schank, Roger C. and Mallory Selfridge, "How to Learn/What to Learn," IJCAI-5, pp. 9-14, 1977.

Selfridge, Mallory, "A Computer Model of Child Language Acquisition," 7th International Joint Conference on Artificial Intelligence, pp. 92-108, 1981.

Sembugamoorthy, V., "PLAS, A Paradigmatic Language System: An Overview," IJCAI-6, pp. 788-790, 1979.

Sembugamoorthy, V., Analogy-based Acquisition of Utterances relating to Temporal Aspects, 1981. Draft. Submitted to IJCAI-7.

Siklossy, Laurent, "A Language-Learning Heuristic Program," Cognitive Psychology, vol. 2, pp. 479-495, 1971.

Siklossy, Laurent, "Natural Language Learning by Computer," in Representation and Meaning: Experiments with Information Processing Systems, ed. H. A. Simon and Laurent Siklossy, pp. 288-328, Prentice-Hall, 1972.

Siskind, Jeffrey M., "Acquiring core meanings of words, represented as Jackendoff-style conceptual structures, from correlated streams of linguistic and non-linguistic input," Proc. 28th Annual Meeting of the Association for Computational Linguistics, pp. 143-156, 1990.

Smadja, Frank A. and Kathleen R. McKeown, "Automatically Extracting and Representing Collocations for Language Generation," Proc. 28th Annual Meeting of the Association for Computational Linguistics, pp. 252-, 1990.

Solomonoff, R., "A new method for discovering the grammars of phrase structure languages," Proc. Int'l Conf. on Information Processing, 1959.

Sparck-Jones, Karen, "Mechanized Semantic Classification," 1961 Conference on Machine Translation and Applied Language Analysis, vol. II, pp. 418-435, Her Majesty's Stationery Office, London, 1961.

Turk, Christopher C. R., "A New Model of NL Acquisition by Machine," draft, 1984.

VanLehn, Kurt and William Ball, "A Version Space Approach to Learning Context-free Grammars," Machine Learning, vol. 2, pp. 39-72, 1987.

Wagner, M., The Application of a Learning Technique for the Identification of Speaker Characteristics in Continuous Speech, Ph.D. Thesis, ANU, Canberra ACT, 1978.

Wirth, R., "Lernverfahren zur Vervollstaendingung von Hornklauselmengen durch inverse Resolution," IWBS Report 84, Wissenschaftliches Zentrum - IWBS, IBM Deutschland, 1989.

Wolff, J. G., "Grammar discovery as data compression," Proc.

AISB-GI Conf. on AI, pp. 375-379, Hamburg FRG, 1978.
Wolff, J. G., "Language acquisition, data compression, and generalization," Language and Communication, vol. 2, pp. 57-89, 1982.
Zernik, U., "Strategies in Language Acquisition: Learning Phrases from Examples in Context," Ph.D. Dissertation, University of California, Los Angeles, 1987.
Zernik, U. and P. Jacobs, "Tagging for Learning: Collecting Thematic Relations from Corpus," Proc. COLING-90, 1990.

### 7.4 Significance

One of the first pieces of work which could claim to be MLNL is that of Yngve, cited in section 3.1. Contemporary is the work of Solomonoff, Spark-Jones and Lamb, with Machine Translation as the primary target, and statistical methods as the primary weapon. This sort of approach was, however, one of the main targets of the theoretical analyses of Gold and Chomsky (section 3) which showed that there were inherent problems with such simplistic approaches. Feigenbaum takes a more AI approach.

Following the example of Yngve, and parallel to the development of corresponding techniques in Connectionism, such techniques are now used in restricted (and normally lower) levels of the language hierarchy and attempt to capture some of the restrictions of human cognition. Such restrictions, as pointed out in section 3 and 4, are somewhat stronger than mere heuristics in that they actually define natural language. This leads to a change of perception, suggesting that similar methods are appropriate at corresponding levels of different modalities (suggesting extension to Ontology), and the different restrictions, teacher-critique characteristics and algorithms may be appropriate for learning at different levels of the hierarchies. Anderson and Powers have built preliminary implementations based on such ideas.

The work of Kelley, McMaster et al., and Harris are also historically significant pieces of work. Harris was one of the first to work with a deep semantics - "the parts of speech are the parts of the robot".

Another important contribution is the recognition of the place of errors - an important source of negative information, on the one hand, and a recognition that language is broader than textbook grammar, on the other. Langley, Powers and Lehman make use of errors rather than cursing them, and Kelley introduced very early the idea of filtering out what did not fit into one's grammar and making use of the borderline, still comprehensible, cases for learning. This approach has been followed also by Reeker. The use of discrimination techniques is related, and essential, and has been pursued by these same researchers.

Some of the work aims to explore the use of a particular technique or approach. Berwick originally started off within Chomsky's TGG paradigm using the Marcus parser. Wirth and Rayner use particular specialized learning techniques in a language context as test bed, and Rayner and Samuelsson have, in particular, achieved impressive improvements in efficiency through the application of their approach. Salveter and Selfridge worked in the context of Schank's Conceptual Dependency Graphs (section 2.2).

Applications have also called some projects into being, Rayner's we've mentioned. Zernik's, Wolff's, Wagner's are also examples of this. Some work has had a very specialized focus, outside of the traditional preoccupation with grammar. Dresher and Kaye are concerned with Phonology, and Granger, Siskind and Sembugamoorthy (building on the approach of Narasimhan) with particular aspects of Semantics. There is a lot more room for work in what should not be peripheral areas.

### 7.5 Goals and Issues

GOAL: The HAL of 2001, or Bridging the Communication Gap?

ISSUE: Most systems, and natural language learning experiments, are in danger of just translating from one representation to another. While this is appropriate for specific applications (database, machine translation, etc.), there is little merit in learning a one to one correspondence, or somethings close to it. Implementers need to make clear they are doing more than that.

ISSUE: Humans learn their language in parallel with their ontology! That is humans have to learn about their world too! A language learning system which cannot learn about its world is not adaptable, and has impaired language learning capability.

ISSUE: Most systems, and natural language learning experiments, start with simple examples of sentences (and/or meanings) and work up (if they're lucky) to complicated examples. Children learn primarily from full blown adult conversation. There is relatively little (machine readable) graded material. There is little advantage in constructing examples by rule. Learning is only possible of "what we almost already know". To use material which is beyond this "next grade" level, we need "filtering" - heuristic elimination of unprocessable input.

ISSUE: Some "field" systems provide mechanisms for accommodating to overly complex or new input, and optimizing to user variation and development. But the "too hard basket" is discarded. This, however, is precisely where learning systems should focus their effort, what is beyond the range of "acceptable" but nonetheless still "understandable". The excess baggage is never gratuitous!

ISSUE: What is the relationship between learning for recognition and learning for production? Children's generation capability seems to lag their understanding. Computers often reverse this trend!

ISSUE: Performance related learning is a factor in language learning, and a precursor to other aspects of language learning. But what role does it have and how can performance related developments in specialized domains incorporate into HALs.

ISSUE: Organization and consolidation have not been problems in some toy systems or specifically applied adaptive contexts. But in general, learning to associate similar things, classify and consolidate, can create problems in relation to memory. Programmers don't like to throw anything away. (It can involve implementational difficulties anyway.) People don't remember everything(?). And they certainly don't remember everything with the same ease or for the same time. Clutter can be a problem. The frame problem is really a special manifestation of this.

METRICS:
1: Who provides the examples? (Teacher)
2: Who corrects the examples? (Critic)
3: Who evaluates the grammar? (Cheat)
4: How is meaning represented externally? (Examples)
5: How is meaning represented internally? (Knowledge)
6: What is the function of the system? (Interaction)
7: What aspects of grammar are learnt? (Phoneme to Book)
8: What aspects of semantics are learnt? (Noun to Article)
9: What aspects of ontology are learnt? (Robot or Database)

These are the metrics I have used in relation to the research listed. A comprehensive tabulation on the basis of such a list of metrics does not yet exist. I make an "impressionist" attempt above. One day,...

#### 7.5.1 Systems Development

None of the contributions describes all singing, all dancing, all understanding Natural Language Learning systems. But all of them represents some progress along the way.

However, some language learning techniques are already promising to make it into the field. Larry Reeker describes progress with Adaptive User Interfaces, Mark Goodman uses adaptive Case-Based Reasoning and Christer Samuelsson and Manny Rayner use Explanation Based Learning to provide impressive efficiency and efficiency improvement in Natural Language Database applications.

### 8. Apolodgements

In conclusion, what more can I say? The field lies open! Who must I acknowledge? I must acknowledge all whose work I have cited here. And those whose work I have misse d out on, misconstrued or undervalued? Please let me know! I'm sure you realize the impossibility of holding in one's head every detail of such a fast expanding and interdisciplinary area, or even every one of the 40 accepted contributions to this symposium.

And I would appreciate it if others with relevant research and interests would contact me. I hope that the above headings, metrics, goals and discussions may be of some help in evaluating and guiding your own contributions to this area.

Finally, I wish to thank the committee members and participants who have contributed their time and their papers, and in particular I wish to single out Larry Reeker for special thanks for his organizational help.

# A Model of Symbol Grounding in a Temporal Environment

Brian T. Bartell * Garrison W. Cottrell

Department of Computer Science and Engineering
University of California, San Diego
La Jolla, Ca. 92093-0114

**Abstract**

Recent work by researchers [Cottrell, et. al. 90] has focused attention on the Symbol Grounding Problem, which can be paraphrased as follows: if the symbols in a symbol processing system have no computable relationship with the objects or constructs they are to denote in the world, how can the symbols have a non-trivial semantics?

We present a recurrent neural network model which learns to generate symbolic categorizations of the temporal characteristics of its stimulus environment. The Symbol Grounding Problem is addressed by relating the learned categories directly to the perceptual input, and by analyzing the representation space constructed by the network to perform the task. We demonstrate that such a grounded system can exhibit useful generalization, and that the internal representation of the symbolic classes is usefully different than the traditional predicate logic approach.

## 1 Introduction

We wish to investigate how a neural network can learn symbolic classifications of data with strongly temporal features, and how these classifications relate to traditional symbolic approaches to class membership. By grounding the system in an analog environment, a semantics can be ascribed to the learned symbolic classifications. This enables us to analyze the representational system with direct reference to the network's environment. A relevant contrast which we will draw exists between the all-or-nothing nature of a logic predicate $P$, which segments the world into all things $P(x)$ and all things $\neg P(x)$, and the graded and textured potential of PDP representations.

This paper presents a simple recurrent neural network (SRN) which was trained to generate sequences of symbols (which may be interpreted as words from a very simple lexicon) classifying sequences of perceptual input originating from an environment. The symbols generated, although from a small set, constitute classifications of an environment which is both analog valued and which has strongly temporal features. Therefore, the network must correctly learn the temporal regularities in order to successfully generalize to novel sequences from the same continuous space.

## References

[Cottrell, et. al. 90] Cottrell, G. W., Bartell, B. T., Haupt, C. *Grounding Meaning in Perception.* German Workshop on Artificial Intelligence (GWAI), 1990.

*Supported by a Cubic Fellowship and by Peregrine Systems, Inc., Carlsbad, Ca.

## 1 Task

The stimulus environment is defined by a (possibly infinite) set of movies presentable to the network, each of which consists of a sequence of visual images in a retinotopic (or some alternate more abstract) representation. Each image is a static snapshot of the world. At each discrete time step, a single image from the current active movie sequence is presented to the network. Only one movie is active at a time. The particular environment used in the current experiments involves movies in which a (billiard) ball rolls around a square table using a starting position and velocity randomly determined for each movie, and bounces off the table's walled edges. A single movie consists of 20 snapshots of the ball in successive positions on the table. An image is presented to the network using 2 nodes, representing the ball's $< x, y >$ position. The table walls are located at $\pm 0.8$ along both the $x$ and $y$ axes. Velocities $\delta x$ and $\delta y$ are randomly chosen in the range $[-0.3, +0.3]$.

Descriptions of the environment are sequential enumerations of temporal features of the movies. In the "billiard ball" world, descriptions are sequences of the form: "rolling { up | down } and { right | left } { slowly | quickly } period", where the correct choice from each pair is instantiated deterministically based on the current trajectory of the ball. One symbol is generated at the Output layer each time step. Note that the net must learn non-trivial temporal features of the image sequences (e.g. relative rate of motion) in order to generate an accurate description, and that these features are not present in any single image. Additionally, since the perceptual input is rational valued, the network cannot simply memorize a finite corpus – an important distinguishing feature between this work and the work of others in the field ([Allen 90]).

Each word is encoded using a local representation across the Output units. Thus, the representation for each word has a single unique node with a value +0.8, and all other nodes with a -0.8 value. During word generation, the unit with the highest activation value determines which word is output by the network.

## 2 Recurrent Neural Network Model

The network architecture is depicted in Figure 1. Because of the task which the network attempts to solve, it is called the Movie Description Network (MDN). In the figure, each labeled rectangle represents a layer of typical connectionist processing units.[1] Arrows pointing generally upward represent uni-directional weighted links which fully connect the source and target layers to propagate activation values, and which are trained using non-temporal back-propagation [Rumelhart 86]. Arrows pointing downward are one-to-one copy links. The MDN architecture is motivated by the work of previous researchers (e.g. [Elman 88] [Allen 90] [StJohn 90]) in connectionist natural language processing.

The network operates as follows: a visual snapshot of the world is presented to the net at the Input units, and the lower portion of the network (the Image SRN) is trained to predict the next visual state (similar to Elman's word prediction task [Elman 88]); at random intervals, the upper portion of the net (the Word SRN) is trained to generate the sequence of words which describes the current world based on the activations propagated forward to the Buffer layer. The Buffer activations remain fixed during the generation of a single description sequence. Word errors are propagated from the Output layer through to the Input layer. Although the primary task is to generate symbolic classification sequences, prediction training in the Word SRN is used to help constrain the information content in the Hidden-1 layer (similar to the "hints" used in [Wiles 90]).

[1] Each node generates an output signal equivalent to a sigmoid, bounded between -1.0 and 1.0, computed on the inner product of the node's weight vector and input activations, plus bias.

## 3 Performance

The MDN was trained for 7 iterations through 50,000 randomly generated movies, using $\alpha = 0.9$ and $\eta = 0.001$ for 2 epochs, $\eta = 0.0005$ for the remaining 5. 20 units were used in each of the internal layers: Hidden-1, Buffer, and Hidden-2.

A summary of the classification performance of the network is provided in Table 1. Each column summarizes performance for a test consisting of 250 randomly generated movies presented to the network, with a single description extracted and analyzed. The "Extended Movie" test summarizes performance when the description is generated after 50 steps of the movie. The high correctness rate indicates that the network has generalized in time, since the network was only trained on length 20 movie sequences. All other tests sampled the description after 7 steps. The third and last columns in the table test for generalization to ball velocities outside of the trained $\delta x, \delta y$ range (i.e. in $\pm 0.3$). Although performance degrades in these cases, it is gradual and follows the same pattern as for the trained movies. Figure 2 depicts locations of "left/right" word errors plotted in the ball's velocity space (for the case $\delta x, \delta y$ in $[-0.9, +0.9]$). Classification errors for the other decision types ("up/down" and "quickly/slowly") are similarly clustered around decision boundaries and show good generalization performance in the central regions of each class.

Note that only semantic class errors are made; the

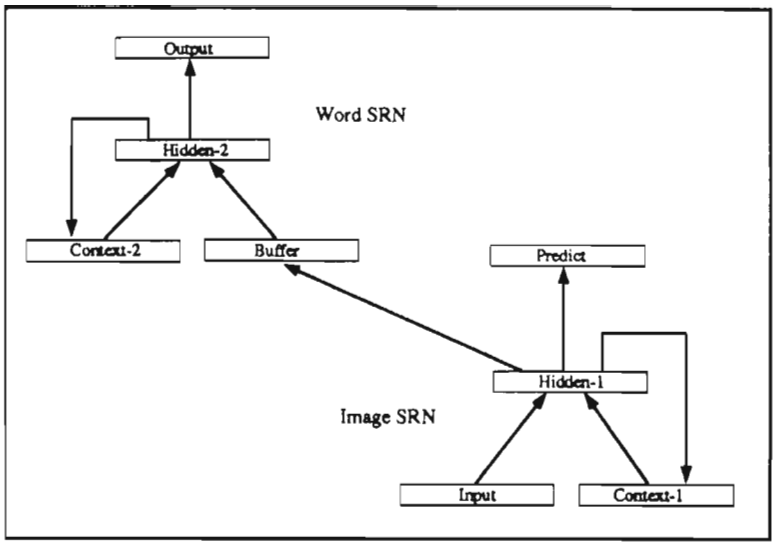



Figure 1: The Movie Description Network (MDN) architecture, described in the text.

| Decision Type | Trained Range | Extended Movie | $\delta x, \delta y$ in $[-0.6, +0.6]$ | $\delta x, \delta y$ in $[-0.9, +0.9]$ |
|---|---|---|---|---|
| Up/Down | 92% | 94% | 89% | 81% |
| Right/Left | 90 | 92 | 87 | 77 |
| Quick/Slow | 84 | 90 | 96 | 96 |
| syntax | 100 | 100 | 100 | 100 |

Table 1: Percent of classifications performed correctly, by decision type, for the trained $\delta x, \delta y$ range and on untrained ranges and movie sequence lengths.

syntax of the description sequences was correct on every test. Classification performance can be improved by further training, although generalization performance for direction classifications degrades more rapidly.

## 4 Discussion

We wish to investigate the character of the trained network's internal representations, with specific emphasis on:

- contrasting the representational space used by the network with traditional binary symbolic classes, and
- visualizing the grounded symbols in terms of the network's environment.

Most of the analysis will consider the Hidden-1 layer, since this layer must encode all information about the environment for the network to perform the primary description and secondary prediction tasks successfully. The Buffer layer is also considered since it encodes class information for the description task.

The first, and most obvious, characteristic of the activations on these layers is that the values are not limited to a small range of values. Rather, the complete range is used. This is expected at the Hidden-1 layer, because of the continuous nature of its prediction task, but also is the case of the Buffer layer. This precludes using a finite state framework for analyzing the recurrence, which is common in the literature [Sun, et. al. 90]. Information is also distributed at the Hidden-1 layer, with node activation values correlated with combinations of the environment parameters $x, y, \delta x$, and $\delta y$. However, the Buffer layer has learned a much more local, although still graded, information encoding, with nodes well correlated with single parameters $\delta x$ and $\delta y$, and with one node correlated specifically with ball speed (a non-trivial feature).

One method for visualizing the grounded symbols is by sampling the network describing a set of movies, and then plotting the classification with respect to the environmental features present. Figure 3 presents this result for the "slow/quick" discrimination. This can be thought of as an approximate extensional semantics for the two symbols, since we define the symbols (e.g. predicate $SLOW()$) with respect to a sample of the

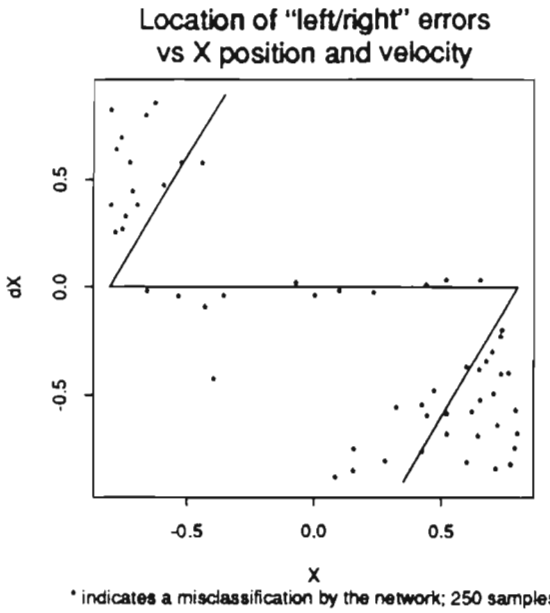



Figure 2: Classification errors occur almost exclusively around the decision boundaries, even during generalization to velocities ($\delta x$) larger than trained ($\pm 0.3$). The horizontal line at $\delta x = 0.0$ denotes the "left/right" boundary. Bouncing (where $x = \pm 0.8$) also separates the classes; diagonal lines indicate regions in which the ball will appear to move one direction (due to sampling the world in discretely timed images) but must be classified as moving in the opposite direction.

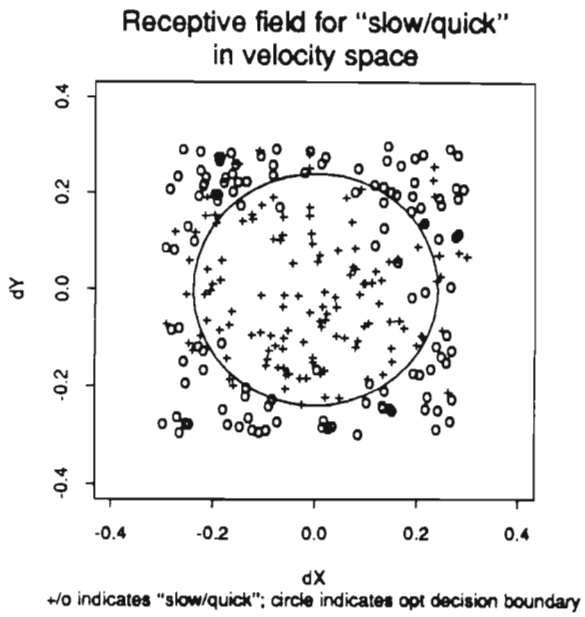



Figure 3: A method for visualizing the meaning of the network's symbols in terms of its grounding environment: the receptive fields for the symbols "slow" and "quick", in terms of sampled ball velocities.

set of things ($X$) which are classified positively ($SLOW(X)$). Note that a single position in velocity space can be classified in two opposite classes at different times, because the recurrent context of the network will be different for different movies.

We may also ask which set of environment feature values is most typical for each class. One process for constructing prototypes for each of the symbols is to average the Hidden-1 activation vectors over a sample which all generated the same symbolic description. These prototype Hidden-1 vectors can be placed in the Image SRN and propagated through the network to examine the network's next image prediction and to check for accurate symbol generation. This process was performed for all atomic decision symbols as well as for a composite: up and right. Symbol generation performance was correct in all cases except for "quick", which generated "slow" instead. Predicted motion for each of the prototypes is displayed in Figure 4, including the up-right composite labeled as "up-right-1" ("up-right-2" is discussed below). Note that the prediction for "quick" is anything but; in fact, the true speed of this prototype is 0.058, well below the 0.240 decision boundary speed.

We may also examine the relationship of other sampled activation vectors to the prototypes. Table 2 lists mean distances (in 20D euclidean space) and standard deviations between the prototypical "up and right" vector and vectors classified by the network in other ways. The increase in distance as the sample vector class moves semantically farther from the prototype indicates that external similarity judgments between representations are possible using a simple euclidean distance metric. However, the large standard deviations suggest interference due to other free dimensions in the system (mainly $x$ and $y$ ball positions, which share the Hidden-1 representation space).

Another technique for constructing prototypes is to use the method of principle components analysis to decompose a very large sample of activation vectors (e.g. Hidden-1) into a set of orthonormal vectors spanning the data space. By letting a base PC vector $\vec{v}$ be equal to middle component values along each axis which we are uncorrelated (empirically) with environment features of interest, and letting the component values of $\vec{v}$ which are correlated with features of interest take on non-mean values, we can calculate a prototype vector $\vec{p} = \vec{v} * R^T$, where $R^T$ is the transpose of the principle component rotation matrix. This method was performed, with prototype "up and right" displayed as "up-right-2" in Figure 4.

| | Up/Right | Right | Down/Right | Down | Down/Left | Left | Up/Left | Up |
|---|---|---|---|---|---|---|---|---|
| Mean | 1.717 | 1.903 | 2.108 | 2.318 | 2.584 | 2.385 | 2.155 | 1.884 |
| SDev | 0.549 | 0.561 | 0.508 | 0.514 | 0.390 | 0.487 | 0.495 | 0.566 |

Table 2: Mean and standard deviation of euclidean distances between the prototypical "up and right" representation and other possible segmentations by class.

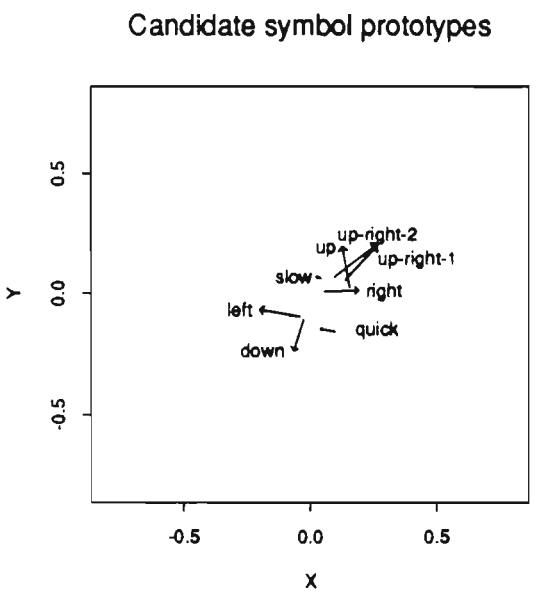



Figure 4: Possible prototypes for each of the classes, including two "up and right" compositions, described in the text.

These two methods for constructing prototypical symbol vectors, and the analysis of distances between representations, assume that the representational space used by the network is essentially euclidean in form. This is certainly explicit in the calculation of euclidean distance, but also is implicit in the averaging operation over representation vectors. By averaging, we assume that symbol representations are clustered in this space. However, it is obvious that these assumptions are not valid in general, taking the calculated "quick" prototype as an example (see Figure 4). It is important to note that the non-linearities in the network (and the additional layers between the Hidden-1 layer and the Output layer) allow arbitrary encodings of the information at Hidden-1, and a convenient euclidean space is therefore not necessarily emergent.

## 5 Conclusion

The Movie Description Network is an architecture for learning a limited descriptive symbolic vocabulary for an analog temporal environment. The labels which the network learns in order to generate correct descriptions of the varied trajectories of a billiard ball appear functionally symbolic, similar to predicate logics; however, the internal representations are distributed and continuous. A Euclidean interpretation for the representation space was examined, and was found to offer insights in general, but was inadequate in at least one case. We have demonstrated that symbols can be grounded in a system's environment, and that such a grounded system is able to generalize to patterns outside its training set.

## References


[Allen 90] Allen, R. B., *Connectionist Language Users.* Tech Report TR-AR-90-402. February 1990. Bell Communications Research.

[Elman 88] Elman, J. L., *Finding Structure in Time.* CRL Technical Report 8801. April 1988. Center for Research in Language, San Diego.

[Sun, et. al. 90] Sun, G. Z., Chen, H. H., Lee, Y. C., and Giles, C. L. *Recurrent Neural Networks, Hidden Markov Models and Stochastic Grammars.* Vol I. IJCNN San Diego, June 1990.

[Rumelhart 86] Rumelhart, D. E., McClelland, J. L., and the PDP Research Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol I.* The MIT Press, Cambridge. 1986.

[StJohn 90] St. John, M. F., *The Story Gestalt: Text Comprehension by Cue-based Constraint Satisfaction.* Preprint, submitted to the 1990 Conference of the Cognitive Science Society.

[Wiles 90] Wiles, J. S., and Bloesch, A. *Patterns of Activation are Operators in Recurrent Networks.* Tech. Report No. 189. University of Queensland, Australia. October 1990.

## From Rules to Principles in Language Acquisition: A View from the Bridge

**Robert C. Berwick, MIT Artificial Intelligence Laboratory**

The central goal of the research program in language acquisition at the MIT AI Lab over the past 8 years has been to build implemented, computational models of language acquisition that can work with *real* databases of parental speech and acquire substantial grammars of different languages. We aim to link grammatical theories and computer models of learning by explicit computer models, while maintaining cognitive fidelity constraints on input complexity, time, and the like. Our ultimate goal, as attested by the associated research of our students and colleagues at this workshop (Brent, Siskind, Clark) is to model in detail *all* aspects of language acquisition: to bootstrap lexical-conceptual and syntactic category knowledge what we know about infants' cognitive capacities; learn word meanings from "reading" unrestricted text; and model later stages of syntax and word acquisition.

In the period 1979-84, these efforts were focused on building an acquisition model that used the then-current "rule-based" representations of syntactic structure and an associated parser. (Berwick, 1979, 1984; 1982 thesis summarized in 1985). Within that framework, several important results were obtained about the constraints required to guarantee learning from positive-only examples (explicit negative examples being assumed cognitively implausible, as is standard in the language acquisition field), and at the same time ensure efficient parsability. The implemented computer model could acquire roughly 100 if-then English-particular rules under a variety of relatively natural positive-only sentence presentations with no prescribed training sequence, at each step yielding an efficient (deterministic) parser. This system worked by incrementally constructing a single new if-then rule based on its inability to parse a novel example sentence. Significant formal results ("learnability theory") included reformulation and application of Gold (1967) and Angluin's (1978) "subset principle" to linguistic examples, as a necessary and sufficient condition on positive-evidence acquisition, and demonstration that efficient parsability implied learnability from simple evidence, in that the constraints that ensured deteterministic parsing also guaranteed learnability. The subset principle was later extensively extended and applied to linguistic examples and psycholinguistic experiments by Wexler and Manzini (1987) and Wexler and Chien (1991).

However, there are large problems with the rule-based theory. First, the system does not allow for errors or retraction of acquired rules; learning was monotonic, in the face of much evidence to the contrary, such as 2-object dative constructions. Second, it was never demonstrated for other languages. Third, the system does not work in the face of ungrammatical or noisy input. Fourth, linguistic theory itself has changed, replacing large numbers of language idiosyncratic rules with a handful declaratively-stated universal constraints ("principles") that vary parametrically over a small range. In this more recent view, acquisition of syntax amounts to setting the parameter values, e.g., whether a language is head-first (like English) or head-final (like Japanese). Our current work is designed to put this newer model to explicit computational test, comparing it to the rule-based view (for a related view comparing principle vs. rule-based approaches, see Fodor and Crain, 1990). To do this, we have constructed the first complete parsing model for a principle-based theory (in the sense of being an efficient parsing system that incorporates the entire range of a modern principle-based theory); see Fong and Berwick, 1989; Berwick and Fong, 1991 forthcoming; Fong, 1991. We can show that by changing just 5 parameters from their English settings we can get a system that handles an interesting subset of the typologically distinct constructions of Japanese, as predicted by the theory. To connect this to acquisition, we are at present using Clark's genetic algorithm approach to learning the parameter settings from unedited motherese as taken from the Childes database (MacWhinney, 1987), using German and English (and eventually, Spanish, Italian, French, etc.) Preliminary experiments indicate that acquisition of full basic tree structures ("Xbar theory") can be accomplished quite readily by using Clark's basic scheme within 150 or so iterations, using unedited text fragments. In fact, these results show that a rule-based system like Fodor and Crain's would work as well as a parameterized system in this domain. Further experiments using actual motherese and a fully parameterized model tied to the parametric parser are underway as this is being written.

### Bibliography for Own Research


Berwick, R., 1979. "Learning Structural Descriptions of Grammar Rules from Examples," Proceedings of the Sixth International Joint Conference on Artificial Intelligence, Tokyo, Japan, pp. 56–58.

Berwick, R., 1984. "Bounded Context Parsing and Easy Learnability," *Proceedings 22nd Annual Meeting of the Association for Computational Linguistics*, Stanford, CA, July, 1984, pp. 20–23.

Berwick, R., 1985. *The Acquisition of Syntactic Knowledge*, Cambridge, MA: MIT Press.

Berwick R. and S. Fong (1991, forthcoming). *Principle-based Parsing.* Cambridge, MA: MIT Press.

Fong, S. and R. Berwick, 1989. "The computational implementation of principle-based parsers," in *Proceedings of the 1st International Workshop on Parsing Technologies*, CMU, pp. 75–84.

Fong, S., 1991. *Computational Properties of Principle-Based Grammatical Theories*, Ph.D. dissertation, MIT.

## 1 Introduction: rule-based acquisition

A central problem for implemented natural language acquisition and machine learning models is the link between parsability and learnability, and the connection between grammatical theories and computer acquisition models.

As a language or a natural grammar is being learned, parsing must evolve hand-in-hand, but obviously one cannot assume that the parser is completely in place. This would assume knowledge of the grammar to be acquired in the first place. On the other hand, if the parser is undeveloped, then example sentences cannot be processed to learn the grammar. To reconcile this paradox, most implemented computer models of syntax acquisition assume an error-driven system: an unparsable example sentence forces an incremental change in the system's existing grammar, typically, the addition or modification of a single, very particular rule. If an example is fully parsable with existing rules then no change occurs.

This method had some success and led to interesting mathematical proofs of convergence (Wexler and Culicover, 1980), working computer models based on specific parsers like the Marcus parser (Berwick, 1979; 1985), and formalized links between easy parsability and learnability (Berwick, 1984; Berwick and Wexler 1987). The Berwick model could acquire 100+ if-then rules as used by the Marcus parser from grammatical (positive) input examples, presented in no particular training sequence. Syntactic category information and an accurate thematic structure for sentences ("who did what to whom") was assumed, as well as most morphological preprocessing these are obviously overly strong constraints that must be relaxed in a more cognitively faithful model (as is being done by our other research group members like Siskind and Brent). The output was a representation of syntactic sentence structure as pictured by a then-current transformational theory along with a thematic (case frame) representation. At each step, the system attempted to parse the input sentence with its current set of if-then grammar rules, in (somewhat modified) Marcus parser form. The *if* portion of the rule was a predicate true of a (approximately) 3-cell input buffer holding words or partially built phrases and the top of a pushdown stack, while the *then* portion of a rule was a single action that could attach one part of a syntactic tree to the top of the stack, create a new phrase, or switch the 1st and 2nd buffer cell contents. If these rules blocked because none applied, or because the output thematic structure did not match that paired with the input sentence, the system would attempt to build a single new rule that would work; if not, it would not process the sentence further at that stage. (This last condition provided a simple kind of simplicity filtering on the input sentences, yielding the presentation order invariance properties of the system.)

This is a simple error-driven model. Nonetheless, the results were still of interest, since they showed that: (1) inference of a full language could proceed on the basis of just simple positive sentences (degree of embedding 2 or less); (2) inference was order-insensitive (it could speed up or slow down depending on the presentation of example sentences, and in fact being faster given the richer percentage of construction types found in motherese); (3) inference was possibly only if one applied a general learning constraint, the *subset principle*, first formulated in a recursive-function theory context by Angluin (1978), such that the system ordered rule hypotheses so that the most narrow language was always guessed first or so that a guessed, possibly overly-general (superset) language always nonoverlapped with the correct (improper) subset target; and (4) inference constraints matched those proposed in Wexler and Culicover's (1980) mathematical model for the acquisition of a transformational grammar, and guaranteed efficient parsability. In particular, the learning model obeyed specific *locality constraints*: e.g., rules could not operate over unbounded domains, nor in some way set up conditions to "hide" possibly incorrect rules over unbounded domains, exactly the Wexler and Culicover conditions for learnability that happened also to ensure bounded context (efficient) parsability. In addition, Berwick (1982, 1985) showed that all of the learning principles then advanced in linguistic theory could be placed under the rubric of the subset principle.

As an example of the subset principle in operation, consider the arguments to verbs, e.g., direct object, indirect object, propositional object. The subset principle would claim that arguments are obligatory until positive evidence is received to indicate the contrary, since the obligatory argument assumption results in a narrower generated language; if the learner first assumed that arguments were optional, and if this assumption were wrong, then no positive evidence could counter this assumption, since the obligatory appearance of arguments would be a subset of the optional argument hypothesis.

**Problems with rule-based acquisition models.** The very nature of such error-driven rule-based methods is their downfall. Since they assume that input will be error-free and consistent, such models cannot readily cope with the actual "ill-formed" or fragment input found in maternal speech. In addition, it is difficult to see how such systems can be extended to different languages: for example, such a system has difficulties in languages like German where simple sentences will appear to be Subject-Verb-Object (via verb movement to the verb second position), but more complex embedded clauses will reveal the "true" verb-final character of German. If the model (or child) receives or can process simple examples first, then this will lead to an assumed SVO order that is violated later on. In general, retraction of hypotheses in such models is difficult. Examination of acquisition envelope curves shows that learning

is "too good": it is monotonic, with no false steps, which is surely not true of actual acquisition. Overgeneralization followed by retraction does occur, it has been argued, for dative verbs in English, among others.

Perhaps most importantly, though, the representation of grammatical knowledge has changed radically in the past decade. Recent shifts in linguistic theory, from language-particular, construction-based rules to declaratively stated universal constraints or "principles", have rendered many older accounts of syntactic natural language acquisition (e.g., Wexler and Culicover 1980) or computer models (e.g., Berwick, 1979; 1982; 1985) obsolete, because these models assumed many language particular if-then type rules. These rule-based models cannot accommodate current linguistic theories that replace many hundreds, even thousands of construction specific rules like passive or dative with a much smaller set of interacting constraints or principles. Our research goal is to see if this change matters for computer acquisition models.

On this more recent view, there is no "rule" of passive, which is epiphenomenal, but rather a set of deductive possibilities arising from more basic axioms that set the basic branching structure of a language's phrases (head initial–English, French; head final–Japanese; mostly head final–German); the direction of case frame or thematic role assignment (left to right in English; right to left in Japanese), and so forth. On this view, acquisition amounts to the setting of parameters given example sentence evidence. While this more modern account accounts for the parametric variation observed across the Romance languages, modern Germanic languages, and some Asian languages, until recently it has had few computer implementations, and these have been incomplete. What has been lacking is a full implementation of a parameterized, principle-based parser and a successful parameter setting algorithm. "Toy" systems have been built that attempt to set a few parameters in a straightforward sequential way, but these do not come close to the full set of perhaps 24 distinct modules each with 3 or 4 parameters to set, and it is not clear that sequential learning will work when scaled up; it is quite easy to get into paradoxical learning sequences where a parameter is first set (as in German), only to have to undo it. (In fact the situation here is far worse than with rules.) In addition, with even, say, 24 modules with 3 or 4 settings, we have 100 parameter values and thus a huge space of possibilities (all the ways of choosing 24 values from 100); this requires too much time and data.

For an explicit computer evaluation, the first step, then, is to a complete principle-based parser; show it can work for multiple languages; and then develop an acquisition model for it. To deal with the multiple-parameter problem, we have chosen to couple our principle-based parser coupling that to Clark's genetic algorithm for parameter acquisition.

The first stage for building a parameter-based acquisition model is to implement a full-fledged principle-based parser that can be parameterized for different languages. This has been met by Fong's implementation of such a parser (in Prolog, see Fong and Berwick, 1989; Fong, 1991) that contains 25 interacting modules parameterized in a few ways each. The parser can successfully cover many hundreds of construction types (not just sentences), in fact, the full range of sentence examples used in a current principle-based linguistic textbook, by Lasnik and Uriagerreka (1988). We will describe how such a parser was readily modified, simply by changing just a few parameters to account for a similar range of Japanese sentences. Thus the parsing system can represent an family of parsers.

We can then turn to the second stage in building a parameterized acquisition model: the parser can be coupled to a simulated evolution (genetic) algorithm to select the right parameters for a given language. The genetic algorithm is designed to be robust against noise, and we have also accommodated the problem of so-called ill-formed input by using Clark's metric of number of structures returned as the metric of 'fitness'. With this metric, even partially well-formed sentences can provide useful input to the learning system.

Let us first describe the parsing model, and then turn to a brief review of the preliminary acquisition experiments that have been carried out by under our direction by our student de Marcken.

## 2 Parsing with parameters: English vs. Japanese

Principle-based language analysis aims to reconstitute the vocabulary of grammatical theory in such a way that constructions like passive *follow* from the deductive interactions of a relatively small set of declarative, conjunctive, node admissibility conditions (the principles). The principles themselves are drawn from the work of that strand of current linguistic theory sometimes called principles-and-parameters theory. How do the principles conspire to replace rules? Space permits only a brief sketch here. For instance, one general principle says that verb phrases in sentences must either *begin* with a verb in some languages, or *end* with a verb in others (those are the degrees of freedom or parameterization in this particular principle). This yields the basic tree shapes in a language, dubbed $\overline{\text{X}}$ *theory*, and gives us part of the variation between languages like English and Japanese. A second principle, called the *Case filter*, says that all pronounced or *lexical* noun phrases like *ice-cream* must receive Case, where Case is roughly an abstract, but universal, version of the Latinate system that gives objective Case to objects, oblique Case to objects of prepositions, nominative Case to sentence subjects, and so forth. Case is assigned either from an active verb like *ate* or an auxiliary verb like *was*; the adjectival form *eaten* does not assign case. A third principle, called the *theta-criterion*, insists that every verb must discharge its *Thematic arguments* and every noun phrase must receive a thematic role, completing a description of 'who did what to whom'. A fourth principle, *Movement* (or *Move-*$\alpha$),

lets one move any phrase $\alpha$ to any available 'landing site'; and so on, for the remaining 21 principles. Now we can conceptually imagine the 'parse' of a sentence to take place via a naive generate-and-test algorithm that, given a PF, enumerates all admissible S-structures, and from there, applies the conjunctive constraints of D-structure and LF to obtain an LF output. For example, given *The ice-cream was eaten*, the system can 'guess' an S-structure such as [$_S$ [$_{NP}$ the ice-cream$_1$ [$_{VP}$ was eaten [$_{NP_1}$ ]]]], that turns out to meet all constraints, without ever positing a passive rule. The system knows from the lexicon that *eat* may assign a thematic role of *Affected Object*. Further, $\overline{X}$ theory says this object must appear to the right of the verb, and Case assignment says that the object appear immediately to the right. However, this cannot be a full lexical noun phrase, because this would violate the Case filter: in the lexicon, *eaten* is an adjectival form that does not assign Case. The only option left is to insert an unpronounced, nonlexical noun phrase (NP) after *eaten*. *Eat* must still discharge its thematic role, to the nonlexical NP, and does so. Now let us check *ice-cream*. It receives nominative Case in Subject position as usual; it can receive a thematic role only if it inherits it from some position, again since *was eaten* is a predicate adjective. Thus the only remaining choice is to link the nonlexical NP after *eaten* to *ice-cream*, which is done by the device of coindexing (subscripting). In practice, this generate-and-test mechanism is obviously inefficient, since the principles that apply to S-structure are but a fraction of those that apply in the overall system. Fong's actual design takes into account a number of important design principles that make the system practicable.

Importantly for acquisition, using the *same* set of principles, but with a different language parameter vector and lexicon, the system can be automatically reconfigured to parse Japanese examples instead (Spanish, German, and other Germanic languages are currently being implemented). *No reprogramming or handcoded rule rewriting is required.* Figure 1 shows the Prolog textual English/Japanese differences plus an excerpt from both lexicons (which contain many hundreds of entries when expanded), to emphasize that just 5 binary switches must be reset to parse Japanese rather than English. We would like to emphasize, however, that the system has not been tested on a full range of Japanese sentences. Rather, a range of *wh* questions and other sentences have been evaluated (Lasnik and Saito, 1984). Nonetheless, these sentences display many of the typological Japanese–English differences: (1) *SOV Language* (Japanese is verb final, more generally, head final); (2) *Scrambling.* Apart from the fact that sentences normally begin with a topic and ends with a verb, the order of other elements in the sentence is relatively free. In particular, direct and indirect objects can be switched, direct (and indirect) objects can appear in front of the subject in sentence-initial position. (3) *Empty subjects.* Subjects (other NPs) can be omitted in Japanese. In general the conditions that determine which elements can or can not be omitted are largely dependent on discourse considerations; (4) *No visible Wh-movement.* In English, in non-echo questions *wh*-words such as *what* must appear in clause-initial position, as in *I know what john bought* rather than *?I know john bought what*; in Japanese, *wh* words appear *in situ*. While obviously this is very far from being a complete characterization of the differences between Japanese and English, it is sufficient to cover a wide variety of *wh* questions, including those in the Lasnik and Saito (1984) article on English and Japanese. For our acquisition standpoint, what is important is that the principle-based parser can capture *all* these distinctions simply by supplying 5 binary parametric differences plus a new lexicon, as shown in figure 1. (These include some rather subtle distinctions, such as, *Taro-ga nani-o te-ni ireta koto-o sonnani okotteru no*, ('What are you so angry about the fact that Taro obtained') vs. **Taro-ga naze sore-o te-ni ireta koto-o sonnani okotteru no* ('Why are you so angry about the fact that Taro obtained it'), which are the *reverse* of the acceptability facts in English. Consider the sentence, *Taro-ga nani-o te-ni ireta koto-o sonnani okotteru no* ('What are you so angry about the fact that Taro obtained') Here the subject of the matrix clause (= *you*) has been omitted. Also, *nani* and *te* ('hand') have been permuted from the canonical order described above—a simple case of scrambling.) The logical form for this sentence should be something along the lines of: for what $x$, *pro* is so angry about [the fact that Taro obtained $x$] Here, *pro* represents the understood subject of *okotteru* ('be angry').

## 3 Genetic algorithms and rule- and parameter-based learning

Linking of this parameter-based parser to Clark's genetic algorithm for acquisition is still underway as this is being written. However, our student de Marcken has carried out some preliminary simulations that use just an X-bar parameter system on random paragraph sets taken from ordinary text (*Wall Street Journal*), and several hundred unedited parental speech samples from the Childes database (English and German; Nina at age 1;11 and Katrin from roughly that age). Note that knowledge of word categories is still assumed, an assumption that we hope to remove by the use of Siskind's model. The principle-based design incorporates some improvements to Clark's model: first, we use a different genetic algorithm, as described by Schaffer in Davis (1987); second, we use a pure partial phrase parser, rather than a full principle-based system. (A third improvement that the principle-based design will offer is that a blocked parse will point to a possibly offending parameter value directly, instead of randomly, which may improve accuracy and convergence.)

An example of the parameter system:

Note: '\+' denotes 'negation as failure', *** & underlined= English–Japanese differences

| English parameters | Japanese parameters | English lexicon |
|---|---|---|
| X-Bar Parameters | | Proper nouns |
| specInitial. | specInitial. | lex(bill,n,[a(-),p(-),agr([3,sg,m])]). |
| specFinal :- \+specInitial. | specFinal :- \+specInitial. | ... |
| *** headInitial. | headFinal. | Verbs |
| *** headFinal :- \+headInitial. | headInitial :- \+headFinal | lex(arrest,v,[morph(arrest,[]),grid([agent],[patient])]). |
| agr(weak). | agr(weak). | lex(arrive,v,[morph(arrive,[]),grid([theme],[])]). |
| Bounding Nodes | | lex(ask,v,[morph(ask,[]),grid([agent],[?proposition])]). |
| boundingNode(i2). | boundingNode(i2). | ... |
| boundingNode(np). | boundingNode(np). | Japanese lexicon: |
| Case Adjacency Parameter | | lex(biru,n,[a(-),p(-),agr([]),grid([],[])]) bill |
| *** caseAdjacency. % holds | :- no caseAdjacency. | lex(doko,n,[a(-),p(-),agr([]),wh,location,grid([],[])]). |
| Move Wh In Syntax Parameter | | (where) ... |
| *** whInSyntax. | :- no whInSyntax. | lex(are,v,[morph(are,[]),subcat(vp$[morph(_,[])],[])]). |
| Pro-Drop Parameter | | lex(irer,v,[morph(irer,[]),grid([agent],[goal,instrument])]). |
| *** :- no proDrop. | proDrop | ... |

Figure 1: The complete parametric differences between English and Japanese needed to incorporate the Lasnik and Saito theory.

```
(defparam A-SPEC (left right)) ;; Q on right or left of ABAR
(defparam ADV-SPEC (left right)) ;; Q on right or left of ADVBAR
(defparam INFL-SPEC (left right)) ;; DP on right or left of IBAR
(defparam COMP-COMP (left right)) ;; IP on right or left of COMP
(defparam INFL-COMP (left right)) ;; VP on right or left of I
(defparam DET-COMP (left right)) ;; NP on right or left of DET
(defparam V-THETA (left right)) ;; args on right or left of V
(defparam ADJ-THETA (left right)) ;; args on right or left of A
(defparam N-THETA (left right)) ;; args on right or left of N
(defparam RELCLAUSE-ADJUN (left right)) ;; CP on right or left of DP
(defparam AP-ADJUN (left right)) ;; AP on right or left of NP
(defparam PP-ADJUN (left right)) ;; PP on right or left of DP, VP
(defparam ADVP-ADJUN (left right)) ;; ADVP on right or left of VP
(defparam P-CASE (left right)) ;
```

In the preliminary experiments, these parameters ground an X-bar system, so in fact, the current results apply equally to a phrase-structure view like Fodor and Crain's (1990), in fact, are an explicit computer modeling test of a part of their proposals. 25 different random sets of parameter settings are created, and used to parse the text. A rating is assigned, namely the number of phrases returned for the whole text; the actual displayed value is the opposite, so a higher number is better). In each iteration, two parent settings are chosen, and a new setting is created by picking randomly between the values of each parent, and with a small probability, randomly assigning a parameter to any value in its range, regardless of parent settings. The new parameter setting takes the place of one of the 25 previous settings, with the proviso that it can not take the place of any setting which produced a rating above the mean rating for all 25 settings. In the sample runs below, 25 settings have been tested before any results are displayed. Then in every iteration the best five parameter settings (from the 25 in the current population) are displayed.

A sample run:

```
(learn)

Iteration 1.
0.  -449.0  [DET-N ARG-V VP-NP COMP-S NP-PP P-NP AP-NBAR SPEC-V V-ADV AP-QP ]
1.  -455.0  [DET-N V-ARG VP-NP S-COMP NP-PP P-NP AP-NBAR V-SPEC V-ADV QP-AP ]
2.  -467.0  [DET-N V-ARG NP-VP COMP-S PP-NP NP-P AP-NBAR SPEC-V V-ADV AP-QP ]
3.  -475.0  [DET-N ARG-V NP-VP COMP-S NP-PP P-NP AP-NBAR V-SPEC ADV-V QP-AP ]
4.  -486.0  [DET-N ARG-V VP-NP S-COMP PP-NP NP-P AP-NBAR V-SPEC ADV-V QP-AP ]

Iteration 2.
0.  -449.0  [DET-N ARG-V VP-NP COMP-S NP-PP P-NP AP-NBAR SPEC-V V-ADV AP-QP ]
1.  -455.0  [DET-N V-ARG VP-NP S-COMP NP-PP P-NP AP-NBAR V-SPEC V-ADV QP-AP ]
2.  -467.0  [DET-N V-ARG NP-VP COMP-S PP-NP NP-P AP-NBAR SPEC-V V-ADV AP-QP ]
3.  -475.0  [DET-N ARG-V NP-VP COMP-S NP-PP P-NP AP-NBAR V-SPEC ADV-V QP-AP ]
```

```
4.  -486.0  [DET-N ARG-V VP-NP S-COMP PP-NP NP-P AP-NBAR V-SPEC ADV-V QP-AP ]

...

Iteration 170.
0.  -417.0  [DET-N V-ARG NP-VP COMP-S NP-PP P-NP AP-NBAR SPEC-V ADV-V QP-AP ]
1.  -421.0  [DET-N V-ARG NP-VP COMP-S NP-PP P-NP AP-NBAR SPEC-V V-ADV QP-AP ]
2.  -427.0  [DET-N ARG-V VP-NP COMP-S NP-PP P-NP AP-NBAR SPEC-V ADV-V QP-AP ]
3.  -428.0  [DET-N V-ARG NP-VP S-COMP NP-PP P-NP AP-NBAR SPEC-V ADV-V QP-AP ]
4.  -428.0  [DET-N V-ARG NP-VP COMP-S NP-PP P-NP AP-NBAR SPEC-V ADV-V AP-QP ]
```

Convergence has proved relatively stable given the initial starting conditions, noise, and text, though it must be stressed that these results are completely preliminary and have not been fully investigated. It remains to see how the system will work with full motherese in English and German, and with a full set of parameters; we aim to test the parameterization of modern Germanic languages proposed by Webelhuth (1989), as well as our running Japanese system. So far at least, it appears that for basic phrase structure, a rule-based and parameter-based system could perform equally well using a genetic algorithm for acquisition, overcoming many of the traditional obstacles such as noise cited earlier.

## 4 General Bibliography


Angluin, D., 1978. "Inductive inference of formal languages from positive data," *Information and Control*, 45, 117–135.

Berwick, R. and K. Wexler, 1987. "Parsing Efficiency, C-Command, and Learnability" in *Studies in the Acquisition of Anaphora*, B. Lust, ed., vol II. Reidel, 45–60.

Davis, L., 1987. *Genetic Algorithms and Simulated Annealing*, Morgan-Kauffman.

Fodor, J.D. and S. Crain, 1990. "Phrase structure parameters," *Linguistics and Philosophy*, 13, 619–660.

Fong, S., 1991a. Type inference and the recovery *Proceedings of the International Workshop on Natural Language and Logic Programming*, Sweden.

Gold, E., 1967. "Language identification in the limit," *Information and Control*, 10, 447–474.

Lasnik, H. and M. Saito, 1984. "On the nature of proper government," *Linguistic Inquiry*,235–289.

Lasnik, H. and J. Uriagereka, 1988. *A Course in GB Syntax*, Cambridge, MA: MIT Press.

MacWhinney, B., 1987. "Childes, a database for child language acquisition," *Computational Linguistics*.

Manzini, M. and K. Wexler, 1987. "Parameters, Binding Theory, and Learnability," *Linguistic Inquiry*, 18, 413-444.

Marcus, M., 1980. *A Theory of Syntactic Recognition for Natural Language*, Cambridge, MA: MIT Press.

Webelhuth, G., 1989. *Syntactic Saturation Phenomena and the Modern Germanic Languages*, Ph.D. dissertation, University of Massachusetts at Amherst.

Wexler, K. and Y. Chien, 1991. "The acquisition of binding," *Language Acquisition*, **3**.

Wexler, K. and P. Culicover, 1980. *Formal Principles of Language Acquisition*, Cambridge, MA: MIT Press.

# Automatically Inferring Dictionaries from Natural Text and Simple Grammar

MICHAEL R. BRENT

I am developing and implementing algorithms for automatic, unsupervised learning of both syntactic and semantic properties of verbs. The goal is to automatically generate full-scale dictionaries of natural languages using large amounts of naturally occurring text as training data. This work provides the first algorithmic, demonstrably effective approach to three longstanding problems in natural language processing, artificial intelligence, and cognitive science.

- For natural language processing it removes the lexical barrier to scalable, high coverage parsers.
- For artificial intelligence it derives some of the fundamental ontological categories people use from the structure of language, so researchers need not rely exclusively on introspection. What's more, it provides a dictionary linking the concepts it derives to words.
- For cognitive science it provides an algorithmic approach to the bootstrapping problem — namely, how does a learner get a sufficient toe-hold on the vocabulary to make use of the input sentences? These learning algorithms start with only a small, finite-state grammar for a fragment of English and a dictionary of some two-hundred "grammatical" words like pronouns, prepositions, and helping-verbs.

Verbs are the best studied and apparently the richest part of language in terms of syntactic features with semantic correlates, so I have concentrated initially on them. In particular, I am focusing on acquisition of the syntactic argument-taking properties of verbs and the semantic classifications they induce. For example, the verb *expect* can take an infinitival clause like "to eat ice-cream" as one of its arguments, whereas the verb *jog* cannot. This contrast is illustrated in following pair of sentences.

(1) a. I expected [$_{NP}$ the man who jogged $_{NP}$] to eat ice-cream
b. I doubted [$_{NP}$ the man who liked to eat ice-cream $_{NP}$]

As a result of the different argument-taking properties of *expect* and *jog*, the infinitival phrase "to eat ice-cream" is associated with *expect*, not with the adjacent verb *jog* in (1a). In (1b), by contrast, the adjacent verb *like* does take infinitives and the earlier verb *doubt* does not. Algorithms for learning this and other argument-taking properties of verbs from untagged text, along with empirical results obtained with them, are described in Brent (1991a).

The work on semantic classification of verbs depends on the data obtained from the syntactic component described above, and hence the semantic work is at a less advanced stage than the syntactic. Further, the evidence bearing on meaning classification tends to require knowing several, if not all of the possible syntactic argument types for each verb. However, one classification depends on only a single syntactic form, the verbs that take as arguments both a direct object and a sentence at once, as in "John told her he was happy." These verbs all have a sense involving communication, like *advise, assure, convince, inform, reassure, remind, tell,* and *warn*, all of which my program identified (Brent, 1991a). There are at least fifty and possibly as many as one-hundred syntactically identifiable semantic classes like these communication verbs.

In addition to the syntactic and semantic classifications induced by argument structure, some interesting classifications are induced by the verbal auxiliary. For example, verbs whose meaning is purely stative tend not to occur with a progressive auxiliary, as in "* Jon is knowing calculus." Brent (1990) describes initial corpus-based research on this semantic cue, and Brent (1991b) describes an implemented classifier for stativity.

Brent (1990) M. Brent. Semantic Classification of Verbs from their Syntactic Contexts: Automated Lexicography with Implications for Child Language Acquisition. In *Proceedings of the 12th Meeting of the Cognitive Science Society.* Cognitive Science Society, 1990.

Brent (1991a) M. Brent. Automatic acquisition of subcategorization frames from untagged, free-text corpora. Under Review for the 1991 Meeting of the ACL. Manuscript available.

Brent (1991b) Semantic Classification of Verbs from their Syntactic Contexts: An Implemented Classifier for Stativity. In *Proceedings of the 5th European ACL Conference.* ACL, 1991.

# Automatic Semantic Classification of Verbs

MICHAEL R. BRENT

MIT AI Lab
545 Technology Square
Cambridge, MA 02139
michael@ai.mit.edu

Until recently there has been little discussion in computational linguistics of how the meanings of words might be learned automatically. It is not clear how a complete specification of word-meaning might be acquired[1], but there is reason to hope that words might be automatically assigned to ontological classes. For example, Hindle (1990) reports work on automatic semantic classification of nouns based on a statistical profile of the verbs with which they appear. This paper describes work on automatically classifying verbs based on specific constraints between their meaning classes and their syntactic privileges of occurrence. For example, verbs that take a direct object and a complement clause, as in "Jon told **me that** Bill is a fool," are verbs of communication (Zwicky, 1970). An implemented system for automatic classification is described, and successful experiments with two examples, including the communication verbs, are discussed. The system uses minimal syntactic knowledge and parsing machinery, ignoring all but the syntactically most simple cases and using statistics to manage the resulting error, as well as other anomalies. The combination of specific linguistic/ontological constraints, limiting to the syntactically simple cases, and analysis of sampling error gives good results in the automatic classification of verbs. This is important for building computational lexica that automatically track the language. It also offers a more concrete picture of how child language learners with little prior knowledge could use such constraints to narrow down the possible meanings of verbs (Gleitman, 1990).

Brent (1990, 1991b) detail a system that classifies verbs into those that describe only states of the world, like *know*, and those that can describe events, like *fix*. The criteria were how often each verb occurs in the progressive and how often it occurs in with rate adverbs like *quickly* and *slowly*.[2] The

[1] But see Siskind, 1990 for current work on learning relatively detailed definitions.

[2] These criteria are discussed in Dowty (1979) where they are attributed to Lakoff (1965).

following examples demonstrate that purely stative verbs are not natural in these two constructions:

Jon is fixing his car
* Jon is knowing calculus
Jon fixes his car quickly
* Jon knows calculus quickly

Data on the frequency of occurrence of each verb in the sample corpus[3] in each construction were analyzed to determine statistically reliable bounds on their true frequency of occurrence in the absence of sampling error. These bounds were then used to automatically classify the verbs. The results were good — when verbs are classified as purely stative if less than 1.2% of their occurrences are in the progressive (with 95% confidence) and no significant proportion are modified by rate adverbs then six verbs are labeled stative: *know, seem, like, want, believe*, and *remain.* These six are true statives. On the other hand the three statives *mean*, *require*, and *understand* are missed. If the criterion is relaxed somewhat to $< 1.35\%$ progressive then these three are labeled stative, but so is *agree*, which has a non-stative sense meaning to voice agreement. This problem might be resolved by a larger corpus which would give tighter bounds on modification of *agree* by rate adverbs, or by distinguishing between occurrences of *agree* with complement clauses (mostly stative sense) and those without (mostly non-stative sense).

The automatic classification of verbs as purely stative depends on the selection of apparently arbitrary cut-off points for frequency in various constructions. However, it may be possible to choose the cut-offs automatically. The distribution of frequencies in the progressive over all the common verbs showed a marked clustering into distinct populations which might be identified by a regression analysis. Once the clusters are identified, independent data on the stativity of a couple of verbs might be sufficient to classify the rest. This learning algorithm is similar to neural-net methods.

The system of Brent (1990, 1991b) used a parser to identify verbs in the progressive and those modified by rate adverbs, but that work is now being duplicated using only simple regular expressions. The regular expressions work for the progressive construction because it is purely local. Their accuracy in determining which verb is modified by a rate adverb is not perfect,

[3]This work was done on the million-word Lancaster/Oslo/Bergen (LOB) corpus, which includes edited text from a variety of sources.

since arbitrarily many words can intervene between the verb and the adverb. However, these constructions are relatively rare and the only cost of ignoring them is a small increase in the size of the input corpus needed for significant results. Such an increase imposes a much smaller burden, both theoretical and practical, than requiring the learner to have powerful grammatical knowledge and parsing machinery.

The regular expression technique has recently been applied to a second ontological class, communication verbs (Brent, 1991a). As noted above all verbs that take a direct object and a complement clause at once, as in "Jon told **her that** he is a happy," have a sense involving communication. My program picked a number of such verbs out of 2.6 million-words of the Wall Street Journal by finding them in the direct-object-plus-complement-clause construction. In general, distinguishing between a complement clause and a relative clause, finding the boundaries of a direct object, and identifying complement clauses not marked by a complementizer require arbitrary parsing. But the cases where the direct object and the subject of the complement clause are both personal pronouns can be described by a very simple regular expression. Although the examples are few, the test is so reliable that even a single instance is useful. Matching the simple regular expression against a two-million word sample of the Wall Street Journal returned the following verbs, in order of number of occurrences: *tell, assure, convince, inform, remind, advise, persuade, reassure, teach, hit, strike.* Unexpectedly, this list contains verbs of realization, as in, "It hit me that I had been played for a fool," as well as verbs of communication. However, the realization verbs may only appear with pleonastic *it* as their subject, whereas the communication verbs may not appear with pleonastic *it.* What had been thought to be one syntactic criterion for one semantic class turns out to be two criteria for two classes. Interestingly, the realization sense of *hit* is not in Webster's Ninth Collegiate Dictionary, although that sense of *strike* is. This is a reminder of the difficulties inherent in relying on subjective, manual lexicography rather than automated, empirical lexicography.

# Acknowledgments

Thanks to Don Hindle, Lila Gleitman, and Jane Grimshaw for useful and encouraging conversations. Thanks also to Mark Liberman and the Penn

Treebank project at the University of Pennsylvania for supplying tagged text. This work was supported in part by National Science Foundation grant DCR-85552543 under a Presidential Young Investigator Award to Professor Robert C. Berwick.

# References

[Dowty, 1979] D. Dowty. *Word Meaning and Montague Grammar.* Synthese Language Library. D. Reidel, Boston, 1979.

[Gleitman, 1990] L. Gleitman. The structural sources of verb meanings. *Language Acquisition*, 1(1):3–56, 1990.

[Hindle, 1990] D. Hindle. Noun classification from predicate argument structures. In *Proceedings of the 28th Annual Meeting of the ACL*, pages 268–275. ACL, 1990.

[Lakoff, 1965] G. Lakoff. *On the Nature of Syntactic Irregularity.* PhD thesis, Indiana University, 1965. Published by Holt, Rinhard, and Winston as *Irregularity in Syntax*, 1970.

[Siskind, 1990] J. Siskind. Acquiring Corre Meanings of Words, Represented as Jackendoff-Style Conceptual Structures, from Correlated Streams of Linguistic and Non-linguistic Input. In *28th Annual Meeting of the ACL*, pages 143–156. ACL, 1990.

[Zwicky, 1970] A. Zwicky. In a Manner of Speaking. *Linguistic Inquiry*, 2:223–233, 1970.

# Title of talk: Learning Complex Syntax Within a Semantic Parser

**Claire Cardie and Wendy Lehnert**
**University of Massachusetts at Amherst**

**Abstract**

Because conceptual analyzers focus on meaning representations more than syntactic structures, these systems tend to avoid syntactically complicated texts. We describe a cognitively plausible mechanism that allows a semantically-oriented parser to systematically understand complex embedded clause constructions. Furthermore, we outline ongoing work on a machine learning component for our parser that uses a case-based approach to automatically acquire this capability.

## Natural Language Processing Research at UMass

A number of semantically-oriented techniques have been devised over the years to address the problems of conceptual sentence analysis. We have implemented a natural language sentence analyzer, CIRCUS, which incorporates a number of well-known techniques from the symbolic information processing tradition along with original techniques based on numerical relaxation. Our basic system architecture supports a stack-controlled mechanism for managing syntactic predictions, as well as modules for handling two fundamentally distinct types of semantic preferences: predictive semantics and data-driven semantics. A marker passing algorithm is used for predictive semantics, and numerical relaxation is used for data-driven semantics. [Lehnert, W.G. 1990]

The multiple architectures of CIRCUS result in a system that is especially well-suited to the task of selective concept extraction. Portions of sentences that are not covered by the available lexicon can be ignored while intelligible fragments are still processed. Complex syntactic structures such as dependent clauses and participial phrases can be processed without the overhead associated with complete parse trees. Because we effectively ignore those parts of a sentence that are not readily understood, we do not have to design recovery techniques for ungrammatical sentences or syntactic constructs that are not recognized by CIRCUS. These features result in a robust approach to text analysis that utilizes variable-depth processing in order to maximize reliability and minimize processing effort.

Our success with CIRCUS brought us a unique opportunity in 1990. That year CIRCUS was selected as one of about a dozen state-of-the-art systems chosen to participate in the third DARPA-sponsered Message Understanding System Evaluation and Message Understanding Conference (MUC-3). This is a competitive performance evaluation of available technology designed to handle selective concept extraction from wire service stories about South American terrorism. Using a development corpus of 1100 texts, each system is first "tuned" for the target domain before the final system evaluations take place. After about 6 months of development effort, participating systems are then evaluated on the basis of 100-200 test texts. We expect to learn a great deal about CIRCUS during the course of this evaluation.

There is a great potential for computational models that integrate traditional symbolic processing with subsymbolic techniques like backpropagation and numerical relaxation. This seems to be especially true in natural language processing, where many problems can be described in terms of complex constraint satisfaction and preferred (as opposed to correct) interpretations. We believe that CIRCUS integrates symbolic and subsymbolic techniques in a manner that optimizes the complementary strengths of both information processing paradigms.

**References:**

Lehnert, W.G. 1990. "Symbolic/Subsymbolic Sentence Analysis: Exploiting the Best of Two Worlds," In *Advances in Connectionist and Neural Computation Theory*, Vol. I. (ed: J. Pollack and J. Barnden). Ablex. (in press) Also available as COINS Technical Report No. 88-99, Department of Computer and Information Science, University of Massachusetts. 1988.

# Learning Syntax Within a Semantic Parser

Claire Cardie and Wendy Lehnert

Department of Computer and Information Science
University of Massachusetts Amherst, MA 01003

## 1 Complex Syntax from a Semantic Perspective

A relatively large class of natural language processsing systems perform conceptual analysis of text (see [Riesbeck 75], [Birnbaum & Selfridge 81], [Riesbeck & Martin 85], [Wilks et al. 85], and [Cullingford 86]). Because these systems focus on meaning representations more than syntactic structures, it is not surprising that parsers of this class have, for the most part, ignored syntactically complicated texts. Unfortunately, without mechanisms for correctly and consistently handling complex syntactic constructions, conceptual analyzers can only achieve limited success in understanding real stories, research papers, discourse, newspaper articles, etc. Consider the following sentences from a recent article in the Boston Globe:

(1) Last week, the FBI said it had identified people linked with four Middle Eastern terrorist groups who already were in this country.

(2) The Arab terrorist groups that do have infrastructures in the US could carry out terrorist attacks here.

(3) FBI agents have approached members of suspected terrorist support groups to let them know the bureau is aware of their presence.

Like the vast majority of sentences in real texts, examples (1) - (3) contain multiple embedded clause constructions. Understanding these constructs is especially difficult because a natural language system must often infer the existence of a missing constituent in the nested clause and associate it with an antecedent phrase from another clause. In (1) , for example, "people" is both the object of "identified" and the missing actor of the nested clause verb "linked"; "Middle Eastern terrorist groups" is both the object of "with" and the actor of the wh-phrase "were"; in (2), "Arab terrorist groups" is the phonetically null actor of "do have" in the subordinate clause as well as the actor of "carry out" in the main clause; and in (3), "FBI agents" is the actor of both the infinitival complement "to let" and the main clause verb "have approached".

People, however, seem to understand syntactically complex sentences without noticeable effort. Recent experiments in psycholinguistics show that human processing of complicated nested clause constructions is quite efficient [Fodor 89] and there is documented evidence that children understand these constructs by the age of ten [Chomsky 69]. In an attempt to model the way people process language, we have developed a mechanism for systematically understanding nested clause constructions within a semantically-oriented parser called CIRCUS [Lehnert 90]. We define a small number of lexically-indexed control kernels (LICKs) for processing embedded clause constructions and allow individual words to selectively trigger the LICK that will correctly handle the current clause. Each LICK parses a single clause into its semantic representation and then returns that representation to surrounding LICKs using constrained conventions for inter-LICK communication.

In addition, we have evaluated the psychological validity of our approach by comparing CIRCUS' processing of embedded clause constructions with recent psycholinguistic studies of the same constructs [Cardie & Lehnert 91]. Based on this evaluation, we conclude that our architecture is a plausible computational model of human processing for nested clause constructions. This adherence to a cognitively plausible architecture allows CIRCUS to achieve robust sentence processing capabilities not found in other semantically-oriented parsers.

We are currently investigating the possibility of using a case-based approach for automatic acquisition of the LICK definitions that interpret embedded clauses. A more detailed specification of the goals of this ongoing work are given in the final section of the paper. The remaining sections provide a brief overview of CIRCUS and the LICK formalism and present an example of the studies used in our psychological evaluation.

## 2 CIRCUS and the LICK Formalism

CIRCUS [Lehnert 90] is a conceptual analyzer that produces a semantic case frame representation of an input sentence using a stack-oriented control for syntactic processing and a marker-passing mechanism for predictive preference semantics.[1].

In the tradition of conceptual analyzers, CIRCUS' syntactic component produces no parse tree of the input and employs no global syntactic grammar. It is based on the McEli parser [Schank & Riesbeck 81] and uses lexically-indexed local syntactic knowledge to segment incoming text into noun phrases, prepositional phrases, and verb phrases. As soon as McEli recognizes a syntactic constituent, that constituent is made available to the predictive semantics module (PSM) that is responsible for making case role assignments. In CIRCUS, this consists of top-down slot-filling of any active semantic case frames subject to the slot's semantic constraints.[2]

Figure 1a, for example, shows the state of CIRCUS after parsing the sentence "Mary saw the boy". McEli

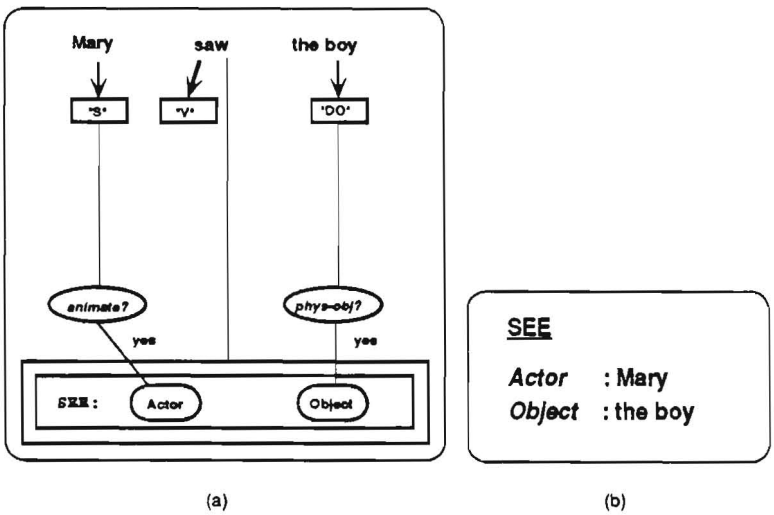



Figure 1: **Mary saw the boy.**

recognizes "Mary" as the subject (*S*), "saw" as the verb (*V*), and "boy" as the direct object (*DO*). In addition, "saw" triggers a semantic case frame for a SEE event. The case frame definition shown in Figure 1a indicates the mapping between surface constituents and case frame slots: subject → Actor and direct object → Object. In addition, it depicts the semantic constraints associated with each slot. Namely, the Actor should be animate and the Object should be a physical object. Because both of these constraints are satisfied, CIRCUS returns the instantiated case frame of Figure 1b at the end of the sentence.

When sentences become more complicated, we have to "partition" the processing in a way that recognizes embedded syntactic structures as well as conceptual dependencies. This is accomplished with lexically-indexed control kernels (LICKs). We view the top-level McEli stack as a single control kernel whose expectations and binding instructions change in response to specific lexical items as we move through the sentence. When we come to a subordinate clause, the top-level kernel creates a subkernel that takes over to process the interior clause. In other words, when a subordinate clause is first encountered, the parent LICK spawns a child LICK, passes control over to the child, and later recovers control from the child when the subordinate clause is completed. Each control kernel essentially creates a new parsing environment with its own set of bindings for the syntactic buffers, its own copy of the main McEli stack, and its own predictive semantics module.

Consider the LICK processing required for the sentence "Mary saw the boy who ran to the lake" (see Figure 2). The top-level LICK is in control until the lexicon entry for "who" indicates that processing of the main clause should be temporarily suspended and a child LICK spawned (see Figure 2a). Because the antecedent for "who" can bind to one of four possible syntactic constituents within the subordinate clause, CIRCUS initializes each of the child *S*, *DO*, *IO*, and *PP* syntactic buffers with "boy". When the child completes a semantic case frame instantiation, at least one of these will be overwritten, and few case frame

[1]CIRCUS also employs a numerical relaxation algorithm to perform bottom-up insertion of unpredicted slots into case frames. This module is not important for the purposes of this paper, however.

[2]CIRCUS allows both *hard* and *soft* slot constraints. A hard constraint is a predicate that *must* be satisfied. In contrast, a soft constraint defines a preference for a slot filler rather than a predicate that blocks slot-filling when it is not satisfied. We will use only soft constraints in the examples that follow.

definitions will reference all four buffers in any case. Figure 2b shows the state of the child LICK at the end

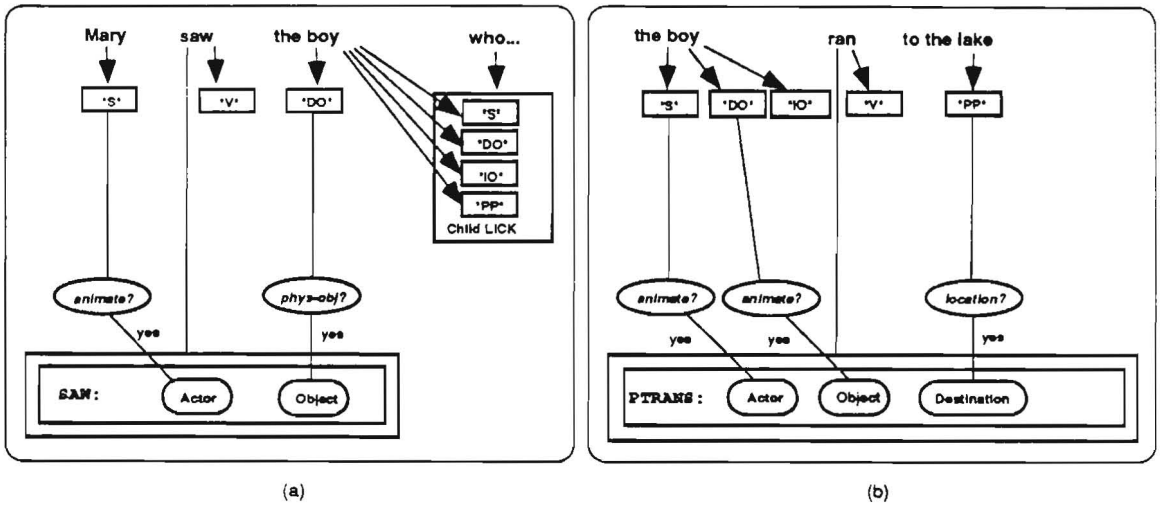



Figure 2: **Mary saw the boy who ran to the lake.**

of the embedded clause. "Lake" has overwritten *PP* and "ran" has triggered a PTRANS[3] case frame. Note that although *IO* still contains the antecedent "boy", it does not interfere with the semantic representation because the PTRANS case frame does not access *IO*. At this point, CIRCUS freezes the PTRANS case frame (with Actor = boy, Object = boy, and Destination = lake), exits the child LICK, and returns control to the main clause where the PTRANS frame is attached to the antecedent "boy".

# 3 Psycholinguistic Studies of Embedded Clause Constructions

Section 2 briefly described how CIRCUS processes embedded clause constructions using its LICK mechanism. In [Cardie & Lehnert 91] we evaluate this approach by comparing CIRCUS to recent experiments in psycholinguistics that address the human processing of nested clauses. In this section, however, we discuss just one of the experiments included in that psychological evaluation.

Consider the following sentence from a Swinney, Ford, Frauenfelder, and Bresnan study:

(1) The policeman saw the boy who the crowd at the party accused # of the crime.

To fully understand this sentence, we have to infer that it is the boy who is being accused — we associate an antecedent or *filler* (in this case "boy") with the missing direct object or *gap* in the wh-phrase (at #). [Swinney et al. 88] determined that people "reactivate" the meaning of a wh-phrase antecedent at the position of its gap in the embedded clause. At # in sentence 1, for example, subjects respond faster to a word semantically related to "boy" (e.g., "girl") than to a control word or to words associated with "policeman" and "crowd".[4] This result implies that people have integrated the meaning of the filler into the current semantic representation of the sentence at the point of the missing constituent. CIRCUS is consistent with this finding. Reactivation occurs in CIRCUS when the next constituent expected according to the McEli stack contains the antecedent. In (1), for example, syntactic knowledge stored with "accused" sets up the McEli stack to expect a direct object to follow. CIRCUS reactivates "boy" immediately following "accused" because the next constituent expected by McEli is the direct object, but *DO* already contains the antecedent "boy".

Furthermore, [Swinney et al. 88] found reactivation only for the correct antecedent at #. They found no reactivation of "crowd" or "policeman". CIRCUS also reactivates *only* the correct antecedent because the LICK formalism makes "boy" the only main clause constituent accessible to the embedded clause. No other noun phrases in the sentence (e.g., "policeman", "crowd", "party") are considered as antecedents of "who".

[3] PTRANS is a primitive act in conceptual dependency describing a physical transfer (see [Schank 75]). The PTRANS case frame actually has a fourth slot — the original location or Source of the object. For the purposes of this example, however, we will ignore this slot.

[4] In the [Swinney et al. 88] study, the target word was briefly flashed at some point during aural presentation of the sentence. Subjects were asked to decide whether or not the visually presented word was a real word and press the appropriate button. Faster response to a target is attributed to priming by the noun with which it is semantically related.

Thus, CIRCUS seems to employ a psychologically valid mechanism for reactivation of antecedents in wh-phrases: it reactivates the antecedent at the point of the gap and it reactivates only the correct antecedent.

# 4 Machine Learning of LICKs

We are currently working on a supervised learning component for CIRCUS that acquires the knowledge encoded in LICKs. For each unique LICK, this component will learn: 1) the lexical items that trigger the LICK, 2) the constituent from the parent LICK (i.e., the antecedent) that should be passed to the child LICK, 3) the set of child LICK syntactic buffers that should inherit the antecedent, and 4) the McEli syntactic predictions that should be in effect at the start of the embedded clause. All of this information is included in the definition of a single LICK.

We plan to use a case-based approach for this language acquisition task where each case consists of 1) the state of the parser (i.e., the McEli stack of syntactic predictions, the contents of the syntactic buffers, the current semantic case frame, the current word, etc.) at the onset of an embedded clause and 2) the desired semantic representation of the embedded clause. Although this machine learning task addresses only a small part of the language acquisition problem, we hope that it offers insights for the development of a more substantial case-based approach to the machine learning of natural language.

**Acknowledgements** This research supported by the Office of Naval Research, under a University Research Initiative Grant, Contract #N00014-86-K-0764, NSF Presidential Young Investigators Award NSFIST-8351863, and the Advanced Research Projects Agency of the Department of Defense which was monitored by the Air Force Office of Scientific Research under Contract No. F49620-88-C-0058. The United States Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon.

# References

[Birnbaum & Selfridge 81] Birnbaum, L. and. Selfridge, M. (1981). Conceptual Analysis of Natural Language, 1981. In R. Schank & C. Riesbeck (Eds.): *Inside Computer Understanding*. Hillsdale, NJ: Lawrence Erlbaum.

[Cullingford 86] Cullingford, R. (1986). *Natural Language Processing*. Totowa, NJ: Rowman & Littlefield.

[Fodor 89] Fodor, J. (1989). Empty Categories in Sentence Processing. *Language and Cognitive Processes, 4*

[Cardie & Lehnert 91] Cardie, C. and Lehnert, W. (1991). A Cognitively Plausible Approach to Understanding Complex Syntax. Submitted to *AAAI-91*.

[Chomsky 69] Chomsky, C. (1969). *The Acquisition of Syntax in Children from 5 to 10*. Cambridge, MIT Press.

[Lehnert 90] Lehnert, W. (1990). Symbolic/Subsymbolic Sentence Analysis: Exploiting the Best of Two Worlds. In: J. Barnden and J. Pollack (Eds.): *Advances in Connectionist and Neural Computation Theory*, Vol. 1. Norwood, NJ: Ablex Publishers.

[Riesbeck 75] Riesbeck, C. (1975). Conceptual Analysis. In R. Schank (Ed.), *Conceptual Information Processing*. Amsterdam: North Holland.

[Riesbeck & Martin 85] Riesbeck, C. and Martin, C. (1985). Direct Memory Access Parsing. In C. Riesbeck and J. Kolodner (Eds.), *Experience, Memory and Reasoning*. Hillsdale, NJ: Lawrence Erlbaum.

[Schank 75] Schank, R. (Ed.) (1975). *Conceptual Information Processing*. Amsterdam: North Holland.

[Schank & Riesbeck 81] R. Schank and C. Riesbeck, 1981. *Inside Computer Understanding: Five Programs Plus Miniatures*. Hillsdale, NJ: Lawrence Erlbaum.

[Swinney et al. 88] Swinney, D., Ford, M., Bresnan, J., and Frauenfelder, U. (1988). Coreference assignment during sentence processing. In M. Macken (Ed.), *Language structure and processing*. Stanford, CA: CSLI.

[Wilks et al. 85] Y. Wilks, X. Huang, and D. Fass, 1985. Syntax, Preference, and Right Attachment. *Proceedings of the 9th International Joint Conference on Artificial Intelligence*. Los Angeles, CA.

# A Computational Model of Parameter Setting

Robin Clark
Département de Linguistique
Faculté des Lettres
Université de Genève
CH-1211 Genève 4
e-mail: clark@uni2a.unige.ch

February 1, 1991

## Abstract

In this paper, I will present a formal model of parameter setting. This model is intended first to demonstrate that a principles and parameters (*P&P*) model of universal grammar (*UG*) has the learnability property and second to provide a formal basis for modelling language acquisition. The learner presented here will correctly hypothesize the parameter settings for the adult target on the basis of exposure to an input text consisting of only positive data; that is, it does not need exposure to ungrammatical strings in order to acquire the syntax of a natural language. Furthermore, it is able to converge to the correct grammar despite the highly ambiguous, equivocal nature of the input data. The system is based on a simple genetic algorithm (Holland, 1975; Goldberg, 1989; Clark, 1990) which exploits natural selection as a basis for learning. The cumulative selectional pressure exerted on the learner over time by the input examples has the effect of gradually pushing the learner to hypothesize the correct target grammar.

The model develops a notion of the *relative fitness* of a parsing device over an input sequence. This metric of fitness is such that (1) the learner can select the best hypothesis from a population even in cases where no available hypothesis can properly account for the input datum; (2) the learner can efficient eliminate inferior hypotheses, allowing for an efficient search of the hypothesis space; (3) successful hypotheses can be compared on the basis of abstract relations (eg, subset/superset relations) so that the most parsimonious grammar can be found. The third property of fitness allows the learner to retract overgeneral "superset" hypotheses (Berwick, 1985) on the basis of positive only input.

## 1 The Learning Problem

In this paper, I will present a formal model of parameter setting in natural language acquisition. Language acquisition is a central problem in psychology, linguistics and cognitive science precisely because it is a case where learners converge to a rich system of knowledge (a grammar) on the basis of highly equivocal, positive-only input data (see Wexler & Culicover, 1980; Morgan, 1986 and the references cited in these works). The system I will present exploits the theory of natural s-

election as a computational framework using *genetic algorithms* (Holland; 1975; Goldberg, 1989; Clark, 1990). In particular, natural language grammars are represented as a sequence of parameter settings which may be taken as a *genotype* which determines a parsing device as its *phenotype*. The parsing devices determined by the learner's hypotheses can be run against the input data to determine their relative fitness in providing well-formed representations for the input, via a fitness metric. The most highly fit hypotheses are then combined to generate new hypotheses which can, in turn, be tested against the input. The system provides a highly efficient means of searching the hypothesis space and is tolerant of ambiguous, relatively uninformative input data. The fitness metric can be defined in such a way as to penalize overgeneral hypotheses allowing the learner to retract hypotheses which generate languages that are supersets of the target without the need for explicit tutoring in the form negative data.

Recent syntactic theory has concentrated on the study of grammatical principles which underlie, and organize, the human natural language faculty. Grammars are organized around a set of universal principles which regulate the assignment of syntactic representations to strings. Language diversity is accounted for by means of a finite set of parameters, vectors along which languages may vary. Comparative syntax and typology can then be viewed as an attempt to determine the variable properties of the human language faculty with respect to the core set of principles.[1] A given parameter may be thought of as a variable inside a grammatical principle which can be instantiated by a value drawn from a finite set of possible values defined by universal grammar.

[1] For a general discussion of this approach to syntactic theory, see Chomsky (1981), Chomsky (1985) and Chomsky (1986).

One well-known example of the interaction between principles and parameters is *subjacency* (Chomsky, 1977), a principle which governs the formation of long-distance extraction, a process that underlies the formation of *wh*-questions (eg, *Who do you think that John saw* e*?* where *who* has been extracted long-distance from the position indicated by *e*). Subjacency forbids long-distance extraction across two *bounding nodes* in a single step. Crucially, languages show a limited degree of variation in the categories that they select to act as bounding nodes, resulting in differential cross-linguistic behavior with respect to long-distance extraction. The task for the learner, then, is to discover which instantiation of the parameters best fits the input data to which it is exposed.

A parameter can be expressed as a simple proposition which may be either true or false:

*IP* is a bounding node.
*CP* is a bounding node.
*NP* is a bounding node.

Given that principles are fixed properties that do not vary across languages, we could specify individual grammars with reference only to particular combinations of parameters values. That is, individual grammars could be represented as strings of truth values (*0* for *false* and *1* for *true*). This representation could then be taken as a way of enumerating the set of possible natural languages in binary numbers. If *UG* consisted of four binary parameters then *1000* (= 8) would be the grammar that results from setting the first parameter to *true* and all the others to *false*. On a more intuitive level, universal grammar may be thought of as a device whose function is regulated by a set of binary switches (the parameters); each switch-setting would determine a parsing device which accepts some natural language.

On this view, the task of the learner is to determine which switch-setting best matches the language it is being exposed to. Formally, then, the learning problem can be described by the following relation:

$$\gamma[\phi_n \circ \varphi(\sigma_i)] = P_m$$

In the above, $\sigma_i$ represents an input text (a sequence of well-formed sentences from the target language $L_i$). The learner is represented by $\varphi$. $\varphi$ produces a sequence of parameter settings (a hypothesis) based on its exposure to $\sigma_i$. This sequence of parameter settings is then interpreted relative to the set of linguistic principles by $\phi_n$ to yield a grammar, $G_i$, for the language from which the input text $\sigma_i$ was drawn. Finally, $\gamma$ maps the grammar produced by $\phi_n \circ \varphi(\sigma_i)$ to a parser, $P_m$, for $L_i$.

Recall that the learner cannot rely on tutoring from negative data. This makes the learning task particularly formidable since the languages generated by different parameter settings may fall into *subset* relations (Berwick, 1985). That is, the language generated by setting a parameter $p_x$ to *0* may be a proper subset of the language generated by setting the parameter to *1*:

$$L[p_1, \ldots, p_{x-1}, p_x(0), p_{x+1}, \ldots, p_z] \subset$$
$$L[p_1, \ldots, p_{x-1}, p_x(1), p_{x+1}, \ldots, p_z]$$

In this case, if the learner overgeneralizes and hypothesizes that $p_x = 1$ when, in fact, the target has $p_x = 0$, the error could be fatal in the sense that no negative evidence will be available to the learner to inform it of its error and all further evidence will be consistent with the learner's hypothesis, which is, after all, a superset of the target language.

A further problem faced by the learner is that several inconsistent parameter settings may derive distinct, but well-formed, representations for the same datum. Suppose that the problematic datum is $s_m$ and that $p_i$, $p_j$ and $p_k$ are parameters such that:

$$s_m \in L_1 = L[p_i(1), p_j(0), p_k(0)]$$
$$s_m \in L_2 = L[p_i(0), p_j(1), p_k(0)]$$
$$s_m \in L_3 = L[p_i(0), p_j(0), p_k(1)]$$

Notice that $L_1 \neq L_2 \neq L_3$. That is, the three hypotheses are not mutually consistent although the grammar for each one derives $s_m$ as a theorem. The learner must have some means, however, of distinguishing between these various hypotheses in the long run.

# 2 Genetic Algorithms

Instead of relying on a costly (and possibly fragile) deductive procedure, Clark (1990) proposes that the causal relation that exists between the input text and hypothesis formation can be most efficiently modeled via natural selection; in particular, Clark (1990) develops a *genetic algorithm* (Holland, 1976; Goldberg, 1989) which models the process of syntactic parameter setting.

In essence, a genetic algorithm consists of the following components:

- A representation of hypotheses in terms of strings, similar in structure to genetic material.
- A measure of fitness of hypotheses in terms of their performance in an environment.
- A reproductive mechanism which allows a hypothesis to produce offspring.
- A Crossover mechanism. This mechanism combines two hypotheses and produces a new hypothesis by combining parts of each to the parent's genetic material.

- Mutation. This mechanism randomly alters an offsprings genotype to produce a new hypothesis close to, but not identical with, the parent's genetic endowment.

The first of the above ingredients is already satisfied by the representation of parameter settings in terms of truth values; the learner's hypotheses can be treated as strings of *0*s and *1*s which have the necessary structure for the other components of the algorithm.

The core component of the algorithm—the one that feeds reproduction and, hence, crossover (the generation of new and better hypotheses)—is the measure of fitness. Intuitively, more fit hypotheses are better at dealing with the problems posed by the input text and, so, should reproduce more prolifically. Thus, the more fit hypotheses will contribute to the formation of new hypotheses via crossover. Gradually, the properties that make hypotheses fit should propagate through the population until the target is converged upon.

I will take parsing as the basis of a measure of the goodness of fit of a hypothesis against the target language. In general, a parser is successful to the degree that it can reduce an input string to a single node in a parse tree; a parse fails if more than one unconnected node is returned by the parser. If two hypotheses fail to parse the input string successfully, we can assume that one is a better hypothesis than the other if the former returns less unconnected nodes than the latter. Finally, as noted above, overgeneral, *superset*, hypotheses should be penalized so that the learner will be able to retract them in light of less general, but still adequate, hypotheses.

These considerations suggest that we can measure the fitness of a hypothesis, $h_i$, and the parser, $p_i$, which it derives relative to a single input string, $s$, from a target language and a population of $n$ hypotheses by means of the following formula:

$$\frac{(\sum_{j=1}^{n} t_j + c\sum_{j=1}^{n} e_j) - (t_i + ce_i)}{(n-1)(\sum_{j=1}^{n} t_j + c\sum_{j=1}^{n} e_j)}$$

In the above, $\sum_{j=1}^{n} t_j$ represents the total number of nodes returned by the population of parsers and $\sum_{j=1}^{n} e_j$ represents the total number of overgeneral parameter settings in the entire population of hypotheses that derive the parsers; $t_i$ and $e_i$ represent the number of nodes returned by the parser $p_i$ and the number of overgeneral parameter settings in $h_i$, respectively. Finally, $c$ is a weighting constant which can be used to fine-tune the relative cost of positing a superset setting for a parameter.

The fewer nodes that an individual parser returns on an input string relative to a population, the more highly fit it will be judged by the above metric. Notice that absolute success in parsing is not a criterion in the above; it is sufficient that a parser returns fewer nodes than its fellows for it to be judged highly fit, but it need not necessarily reduce the input string to a single node. Since the most fit hypotheses reproduce more prolifically and, hence, are more likely to contribute to the formation of new hypotheses via the crossover and mutation operations, the parameter settings that made these hypotheses fit will propagate through the entire population. The inverse of the coin is that less fit hypotheses will tend to die off and, thus, the parameter settings that made these hypotheses relatively unfit will disappear from the population and become unavailable.[2] Furthermore, overgeneral hypotheses will be less robust, allowing the learner to retract overgeneralizations.

[2] The least fit hypotheses are removed from the hypothesis stack with a probability of $p < 0.05$ in the current implementation.

## 3 Parameter Expression

Any given sentence from an arbitrarily selected natural language will be such that it expresses some subset of the parameter settings that go in to making the grammar for that language. That is, the sentence can be successfully parsed by any grammar with the relevant parameters set in the proper way. Other parameter settings will be irrelevant for that sentence. This is just to say that there is a relation between any natural language sentence and the set of grammars which could in principle assign a well-formed syntactic parse tree to that sentence; any one sentence will be compatible with a set of parameter settings.[3] An input datum which expresses some set of parameter settings are, then, *triggers* for those parameters.

Clark (1990) proposes that the set of grammars compatible with a given input sentence can be labelled by virtue of an encoding which enumerates those parameter settings that are necessary to assign a well-formed representation to a given sentence, $s_m$. Supposing that the parameter space consisted of five binary parameters, the p-encoding $\psi$ for $s_m$ might be:

$$\psi(s_m) = [*00*1]$$

where '*' is a variable ranging of *0* and *1*. The above encoding indicates that the sentence $s_m$ can be parsed by any grammar where the second and third parameters are set to *0* and the fifth parameter is set to *1*. Thus:

$$\psi(s_m) = \{00001,\ 10001,\ 00011,\ 10011\}$$

[3] Here, we abstract away from the form of the particular lexical items in the sentence. As observed by Wexler & Culicover (1980), the problem of lexical acquisition can be segregated to a separate learning module, allowing us to consider the mathematical structure of the syntactic acquisition problem.

The target grammar is derived from the application of the learner, $\phi_n$, to the intersection of all the encodings for each sentence in the input text, $\sigma_i$. We presuppose here, as seems natural, that each target parameter setting is expressed by some encoding in $\sigma_i$; an adequate input text must exemplify all those features of the target that the learner must acquire. Note that parsing can be simulated formally be replacing individual sentences in the input text by their encodings and using a simple arithmetic procedure to estimate the success of a hypothesis relative to a given encoding.

The picture that emerges from the above simulation is that the learner is given extremely ambiguous, vague information about the nature of the target. The learner has no direct access to the target parameter settings, only indirect evidence via failed parses on an input text that consists of only well-formed sentences. Given the ability of the fitness function to discriminate between competing hypotheses as well as the inherent robustness of cumulative selection as reflected in the interaction between fitness and reproduction, the current model can successfully converge in large hypothesis spaces despite the extreme poverty of the input data. To date, the model has been tested on a space of 30 binary parameters represented a hypothesis space of $2^{30}$ (= 1,073,741,824) possible languages and has successfully converged in that space.

The reason for the learner's high degree of fault-tolerance is the way in which it exploits the cumulative nature of natural selection to search the hypothesis space for the target. In general, better hypotheses are judged more fit by the fitness metric, reproduce more prolifically and, thus, propagate their beneficial features throughout the population of hypotheses. This, combined with the mutation operation,[4] allow for a highly efficient and ro-

[4] The probability of mutation in the model is cur-

bust learning procedure. By exploiting natural selection, the learner can simulate intelligent design without the exorbitant cost, and brittleness, of deductive procedures, just as is the case in the natural world.

# 4 Summary

The model of parameter setting in the acquisition of natural language syntax presented here presents a learner that is able to converge in a large hypothesis space despite extremely impoverished data. It thus provides an interesting case study from the point of view of engineering a robust, fault-tolerant learning system.

The system makes a number of interesting conceptual and empirical points when considered from the viewpoint of theoretical comparative linguistics, psychology and cognitive science. The ability of the human organism to acquire a first language quickly and efficiently is a remarkable feature of the natural world and the study of this ability stands at the heart of much research in these fields. This approach to language learnability implies that typological analysis of natural languages, empirical case studies of first language acquisition and the theory of parameter setting are all of a natural kind with genetic analysis. Finally, the study of the relationship between language learnability and natural selection promises to provide a strong formal foundation for the notion of a parameter in linguistic theory; if this work is on the right track, then the study of variability within the population of natural language is of the same kind as the study of variability within a population of organisms.

rently set at 0.005.

# 5 References


Berwick, R. (1985). *The Acquisition of Syntactic Knowledge.* The MIT Press, Cambridge, MA.

Chomsky, N. (1977). "On Wh-Movement" in P. Culicover, T. Wasow & A. Akmajian (eds) *Formal Syntax.* Academic Press Inc., New York.

Chomsky, N. (1981). "Principles and Parameters in Syntactic Theory" in N. Hornstein & D. Lightfoot (eds) *Explanation in Linguistics: The Logical Problem of Language Acquisition.* Longman, London.

Chomsky, N. (1985). *Knowledge of Language.* Praeger Publications, New York.

Chomsky, N. (1986). *Barriers.* The MIT Press, Cambridge, MA.

Clark, R. (1990). *Papers on Learnability and Natural Selection.* Technical Reports in Formal and Computational Linguistics, No. 1. Université de Genève.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimisation, and Machine Learning.* Addison-Wesley Publishing Company, Reading, MA.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems.* The University of Michigan Press, Ann Arbor, MI.

Morgan, J. (1986). *From Simple Input to Complex Grammar.* The MIT Press, Cambridge, MA.

Wexler, K. & P. Culicover (1980). *Formal Principles of Language Acquisition.* The MIT Press, Cambridge, MA.

# Applications of Training Data in Semantic Processing

**Deborah A. Dahl**
**Unisys Center for Advanced Information Technology***

February 7, 1991

## 1 Introduction to Research and Bibliography

The problem addressed in this paper is automatic aquisition of the lexical semantics of unknown predicates in natural language processing, based on a quantitative analysis of corpora. This work is being done in the context of the development of Pundit, a large, modular, natural language processing system. The author's particular interests are in the areas of semantic and pragmatic processing and evaluation of natural language systems.

Deborah A. Dahl and Catherine N. Ball."Reference resolution in Pundit", In Logic and logic grammars for language processing, edited by P. Saint-Dizier and S. Szpakowicz, Ellis Horwood Limited, 1990.

Marcia C. Linebarger, Deborah A. Dahl, Lynette Hirschman and Rebecca J. Passonneau, "Sentence Fragments Regular Structures", Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics, Buffalo, NY, June 1988.

Lewis M. Norton, Deborah A. Dahl, Donald P. McKay, Lynette Hirschman, Marcia C. Linebarger, David Magerman and Catherine N. Ball, "Management and Evaluation of Interactive Dialog in the Air Travel Domain", Proceedings of the Darpa Speech and Language Workshop, Morgan Kaufmann, 1990.

## 2 Abstract

This paper discusses two experiments in the application of statistical semantic information in the Unisys spoken language system. The first experiment investigated improving the processing speed in semantics by applying semantics rules in an order reflecting their frequency of application in a training corpus. The second experiment investigated using the training data to make informed guesses about the semantics of unknown predicates. The application discussed here is a database interface to database of information on air travel, such as flight schedules, airfares and ground transportation.

*P.O. Box 517, Paoli PA 19139, dahl@prc.unisys.com, This work was supported by DARPA contract N000014-89-C0171, administered by the Office of Naval Research

# 3 Introduction

Using probabilistic information during natural language processing is a promising means of increasing parsing accuracy, improving processing times, and coping with previously unseen material. Much work has been done in the area of probabilistic parsing ([2], [1]. However, in order for improvements in the ability to provide a parse for new material to have an effect on overall system performance, the rest of the system must also be able to cope with new material. This research describes a technique for allowing the system to make informed guesses about the semantics of new verbs, based on training data. Although there has been some previous work on inferring the semantics of unknown verbs ([5], [4]) this previous work has not exploited the quantitative properties of corpora.

# 4 Experiments in Semantic Training

## 4.1 The Pundit Semantic Interpreter

The semantic interpreter of the Unisys system interprets three types of declarative semantics rules – case frames, rules specifying the mapping of syntactic constituents to the roles of case frames, and rules specifying semantic class restrictions on the fillers of roles in case frames ([3]).

## 4.2 Improving Processing Time

Based on a training corpus of 1000 Air Travel Planning (ATIS) sentences, we have developed by hand an initial set of 200 case frames, 600 syntax/semantics mapping rules and 500 semantic class restrictions. The frequency of occurrence of each of these rules was measured by processing the entire training corpus and recording each successful rule application. The system was then configured to apply the rules in order of their frequency of application in the training corpus when more than one rule could apply. Processing times between the two cases were then compared.

It was found that the only effect of using rules in the order of their frequency was to improve the selection of case frames for polysemous words. Otherwise the semantic search is fairly deterministic, and consequently no speedup through reordering of rules was found. We would expect to see more of an effect in a broader domain with many polysemous words.

## 4.3 Hypothesizing the Semantics of Unknown Words

A second use of this training data is to enable the system to make informed guesses about the case frame structure of new predicates. Previously, semantic processing would fail altogether if there was no case frame for a verb or predicate adjective, thus the system was not very robust when confronted with new words. We wished to provide the system with some means of making intelligent guesses about missing information. We believe that techniques for intelligent guessing should be based as much as possible on quantitative analysis of corpora rather than on hand built heuristics, which can be very effective, but which are difficult to generalize across domains and across languages. For this reason we have developed a guessing mechanism using the frequency

data collected for the experiment described above. Specifically, the frequency of syntax/semantics mapping rules in the training corpus is used to infer likely case roles, given a set of syntactic arguments. For example, the most frequent syntax/semantics mapping in the Unisys ATIS system is a mapping from the syntactic direct object to the theme of the predicate. Consequently, given an unknown predicate with a syntactic direct object, the system will guess that the predicate has a theme and that the direct object maps to the theme role. Other common mapping rules map the syntactic subject to the actor role, 'from' prepositional phrases to the source role, and so on, so for example the presence of a 'from' prepositional phrase in the parse will justify positing a 'source' role in the case frame.

The system can guess case frames in either of two modes. In the supervised mode the guessed case frames are presented to the user in an order reflecting their frequency in the training data. If the user rejects a proposed case frame a less frequent mapping for one of the roles will be selected and new case frames will be generated sequentially until the user accepts one of them. In the unsupervised mode the first guess is assumed to be correct and is used in the current analysis and is output to a file.

One interesting feature of this approach is that the newly guessed case frame is not assumed to represent the complete correct semantics of the verb. Since many verbs have optional arguments as well as several ways of expressing their arguments syntactically, it would be incorrect to simply assume that all the necessary information for the semantics of a verb is given by one instance. In the current system this results in a new guess for each instance of a verb in a corpus. A future improvement to this approach would be to use the algorithm described by [4] in order to incrementally acquire the complete semantics of a verb given a succession of instantiated case frames.

## 4.4 Evaluation

We tested this approach by running 500 ATIS sentences which the system had not previously trained on, while turning on the guessing feature. These sentences contained 8 verbs for which the case frames were guessed. In order to assess the semantic correctness of the guessed case frames, case frames for these predicates were also built by hand and were compared to the guessed case frames. The results, although preliminary, are very encouraging. In general the difference between the linguist's rules and the guessed rules can be characterized by (1) more generality in the hand generated rules, covering anticipated examples beyond the specific utterance containing the verb in question and (2) recognition of synonymy relationships between new and old verbs. We believe that differences (1) and (2) will tend to level out as the system receives additional training data. Difference (3), recognition of synonymy, is an independent issue from inference of argument structure, and will require a different treatment. In order to measure the increase in the robustness of the system provided by this technique we also ran 195 previously unseen ATIS sentences and found that semantics failures were reduced from ten percent to eight percent of all queries. Unfortunately the small number of verbs involved makes generalization difficult, so we plan to repeat these experiments with additional data as it becomes available.

# 5 Future Directions

Because semantic correctness of case frames is not sufficient for system accuracy (the case frame must mean something to the application) an important next step in this research will be to investigate ways of automatically making the case frame meaningful to the application. To do this it will be necessary to determine where the new predicate belongs in the knowledge base. We plan to explore using on-line knowledge sources such as thesauri to address this problem. Once the position of the new predicate in the knowledge base is determined, the application component can use its knowledge of what has been done with semantically similar verbs to decide what should be done with the new verb.

Another important aspect of the meaning of verbs is what kinds of entities act as their arguments. For example *think* requires that its actor be a human. These requirements are needed because they provide an important source of constraint on the analysis, allowing the system to penalize potential analyses which violate them. In current systems these constraints are added by hand, which can be time consuming, prone to inaccuracies, and requires a trained specialist. We will explore automating this aspect of natural language processing by allowing the system to process a corpus while assuming that whatever type of entity appears to fill a role is correct and then aggregating the types. For example if only humans and animals appear as role fillers for some role of a verb then the system may be able to assume that this role requires an animate filler.

# References

[1] Mahesh V. Chitrao and Ralph Grishman. Statistical parsing of messages. In *Proceedings of the Speech and Natural Language Workshop.* Darpa, Jun 1990. Session 8.

[2] David M. Magerman and Mitchell P. Marcus. Parsing a natural language using mutual information statistics. In *Proceedings of AAAI 90*, pages 984–989. AAAI, 1990.

[3] Martha Palmer. *Semantic Processing for Finite Domains.* Cambridge University Press, Cambridge, England, 1990.

[4] M. Webster and M. Marcus. Automatic acquistion of the lexical semantics of verbs from sentence frames. In *Proceedings of the 27th Meeting of the ACL*, Vancouver, 1989.

[5] Robert Wilensky. Extending the lexicon by exploiting subregularities. In *Proceedings of the Darpa Speech and Natural Language Workshop.* Morgan Kaufmann, June 1990.

T MARK ELLISON
Department of Computer Science
University of Western Australia
Nedlands, W.A. 6009
Australia
*marke@bison.cs.uwa.oz.au*

### Research Interests

My research domain is the machine learning of phonology. The aim is to produce programs which, when faced with a collection of phonological data, will abstract symbolic rules and/or representations of the data. In this sense, the research tries to model a linguist, rather than a naive language learner. To date, the work has concentrated on three learning tasks. The first task, acquiring planar segregation, will be discussed in my talk here. The second is the task of acquiring a model of a harmony system given sequences of harmonising vowels. The third and current project involves the acquisition of syllable structure and sonority hierarchies.

In characterising my research, the approach is as significant as the domain. Underlying the approach is the view that a learning system does not *need* domain-specific information. This view affects the style of the learning systems. Asssumptions such as '*consonants are more frequently intial than vowels*' are not permitted. No ad-hoc information about the segments in the wordlists (the data) is permitted: an 'a' can only differs from a 'b' contextually. So the data is purely structural. Finally, the algorithms used are not domain-specific. A domain-independent measure of simplicity is applied to all hypotheses that are compatible with the data. This set of hypotheses is then searched with a domain-independent strategy to find the simplest hypothesis, which is returned as the answer.

### Bibliography

Ellison, TM (1990). **Discovering Planar Segregations**, University of Western Australia, Department of Computer Science Technical Report 90/5.
Ellison, TM (1990). **Discovering Rules for Vowel Harmony**, Proceedings of the 1990 Department Research Conference, Department of Computer Science, University of Western Australia.

### Abstract of Talk

## Discovering Planar Segregations

People can learn languages quickly from little information, defying the complexity of the task. One common explanation states that the language learner has a detailed domain-specific model of language built in, and that language learning is only setting parameters to this model. It is possible, however, to learn about a language with little or no a priori information. In this talk I present an algorithm for finding planar segregations, such as discussed by McCarthy (1989), of phonemes for particular languages. This algorithm requires no domain-specific knowledge of phonology or phonetics. Despite this lack of knowledge, the implemented algorithm has identified structurally significant segregations for thirty languages.

Language acquisition is a problem. The language learner, faced with a finite corpus of moderate size selects consistently from a very large number of possible grammars. The problem is two-fold: (a) by what criteria is the choice of grammar made, and (b) how is the choice computationally feasible? Computational theories of acquisition have a considerable advantage in answering these two questions. If an implementation of a particular theory (a) works, finding correct grammars or grammar fragments, and (b) does so within a reasonable time, then this constitutes experimental verification that the theory has answered the acquisition problem for that grammar or grammar-fragment. Non-computational theories cannot achieve this level of verification.

In this paper, I present an implemented algorithm for finding phonemic planar segregations, using a model of segregation similar to that described by McCarthy (1989). In keeping with almost all models of planar segregation (McCarthy (1981,1989) Prince (1987)), the segregations found by the implementation of the algorithm consistently segregate phonemes onto two planes: one of consonants, the other of vowels.

This result leads to the central thesis of this paper in two steps. First, I examine two competing criteria for how the choice of grammar fragments is made: (a) using a domain-specific learning function, or (b) using a domain-independent learning function. Domain-independence is a priori preferable, by Occam's Razor. A domain-specific model of learning should be assumed only if a domain-independent one is not possible, that is, if choosing a grammar fragment without specific knowledge is not computationally feasible. Second, the program which learns planar segregation is argued to be domain-independent. The main thesis follows: language learners do not need domain-specific information in order to discover planar segregations.

# 1 Acquisition and Learning

**Special nativism.** The acquisition problem may be tackled by denial. There are not really a large number of possible grammars from which the language learner can choose. Rather the language learner has at hand a significant amount of a priori information in the form of a universal grammar which constrains the choice of grammar. The class of grammars which must be investigated is small, and there may only be one possible grammar compatible with the corpus. Following O'Grady (1987) I shall call this approach to the learning problem *special nativism.*

**Problems.** There are two problems with special nativism. Occam's Razor, in one interpretation, forbids us from making unnecessary assumptions. The burden of proof rests with the special nativist to show that each part of the innate universal grammar is essential. If it can be done without, then it should be.

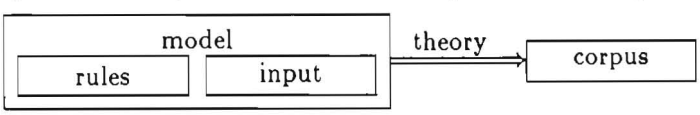
The second problem with special nativism is that it argues for a specific learning function devoted to language and separate from learning systems in other domains, hence its name special nativism. Once again, Occam's Razor enjoins us to avoid such domain specificity unless it is necessary.

These are by no means problems with universal grammar per se, or its discovery as a scientific goal of linguistics. Rather these are problems for the view that language learning is governed and directed by an innate knowledge of universal grammar.

**Minimalist learning.** An alternative approach argues that while the problem of grammar choice might be hard, it is by no means intractable. Rather than assuming as much as possible in an effort to minimise the number of possible grammars, the number of assumptions in the theory is minimised. I shall term this approach to grammar choice the *minimalist* approach. Effort is then oriented towards finding effective search strategies which can deal with the large number of possible grammars. The search strategy should be as independent of the search space as possible.

A weakly restricted class of grammars will usually offer a large number of compatible grammars. An evaluation measure is used to select between grammars not distinguished by the corpus. Just as with the other parts of the learning system, the evaluation measure should be as free from domain-specific assumptions as possible. A very general evaluation measure, perhaps the most general possible, is that proposed by Solomonoff.

**Solomonoff induction.** Solomonoff (1964) proposed an evaluation measure which is applicable to finding models of data under any computable theory. Theories are regarded as idealised computers, on which rules, such as grammars, run as programs that may access some data files, and in any case produce output. Rules plus any required input which produces a particular corpus in the context of a particular theory is called a model of that corpus.




The Solomonoff evaluation measure assigns to each model its size in binary bits. The best of a collection of models which produce the same corpus is the smallest. If we have a 200K program and a 2K program which can produce our corpus from the same input, say 25K, then the 2K program is better: 200K + 25K = 225K > 27K =

2K + 25K. On the other hand, if these programs require inputs of 5K and 500K respectively to produce the corpus, then the first model is better: 200K + 5K = 205K < 502K = 2K + 500K.

The Solomonoff evaluation measure is not specifically oriented towards any domain of knowledge. Under any computable theory and for any collection of data, the evaluation measure can be used to decide between competing models.

Where the data consists of strings of symbols, and no contextual dependencies occur within the strings, it can be shown that the length of the best encoding of a string, $s$, is given by $k + E(s)$ where $k$ is a prefix indicating the encoding of the string, and $E(s)$ is the Shannon entropy of the string. In all the cases considered here, $k$ will be independent of the $s$, and so can and will be ignored. Given the number of times $n_i$ that each symbol $i$ occurs in a string $s$ of length $n$ over the alphabet $A$, the Shannon entropy of $s$ can be calculated by

$$E(s) = n \log_2 n - \sum_{i \in A} n_i \log_2 n_i$$

# 2 Planar segregation

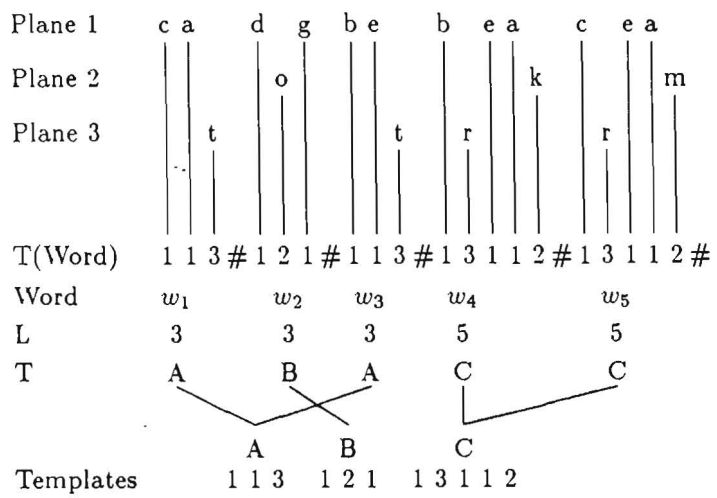
**The Model** McCarthy (1989) describes a theory of planar segregation which provides a good basis for a simple, computable theory. In his model, each word is represented, not as a single linear sequence, but rather as a number of planes each composed of phonemes from a particular class, together with a **template** which describes how these planes are interleaved to form the word. In the most commonly proposed models there are two planes: one for consonants and one for vowels (McCarthy (1989), Prince (1987)).

The model of segregation that is used here is essentially the same as McCarthy's. The major difference is that, rather than hold a copy of the template with each word, I replace these copies by a function which assigns to each word the appropriate template.

As an example, suppose the segment inventory of our language is the English alphabet, and this is divided into three classes: a-g, h-q, r-z. A corpus containing the words 'cat', 'dog', 'bet', 'break' and 'cream' is represented by the planes 'cadgbebeacea', 'okm' and 'ttrr', together with the templates '11312113112' where the digits specify the plane from which the next element is to be taken. It is worth noting that there are no symbols to mark the end of a template.

The length of templates, and hence, how to divide the template string into the individual templates, can be deduced from one observation and two functions. The observation is that the length of a template must be the same as the length of any word that uses that template. The first function $L$ from words onto natural numbers, states the length of each word. The other function, $T$, is very important in reconstructing the corpus from the model. It indicates which template is to be used for each word. To determine the length of any template, find a word which is mapped onto that template by $T$. The length of that word, as given by $L$, must equal the length of the template. Taking each template in turn, we can use this length information to insert separators between templates. Because end-of-template markers would be redundant, they need not be used within the model.

In our example, the function $T$ is 'ABACC', where A is the first template, B the second and C the third. The function $L$ shows the lengths 33355. From these we can deduce the length of the templates: 335.



From these three components of a model: the planes, the templates and the functions assigning a template and a length to each word, the corpus may be derived. *If the classes of segments assigned to particular planes are disjoint* then there is a one-to-one correspondence between classifications of segments and models. For the sake of simplicity, and to reduce the size of the search space, it is assumed that the classes are disjoint.

**Evaluating planar segregations** The Solomonoff evaluation of a planar segregation model is the sum of the length of encoding as strings: (i) each plane, (ii) the template list, (iii) the functions $T$ and $L$, template choice and word length respectively. The length of the encodings is just the Shannon entropy of the strings, when the probability of occurrence of any symbol in the string is just the relative frequency of its occurrence.

The word-length function, $L$, depends only on the corpus and not on the segregation. No matter how it is encoded, it has no impact on the choice of segregation, and so does not need to be considered when evaluating segregations.

Now let us look at the example of planar segregation given in the last section, and evaluate it component by component. The first plane is the string **cadgbebeacea**. The absolute frequencies of the six symbols, **abcdeg**, which occur in this string are 322131, and the length of the string is 12. The Shannon information measure of the string is

$$29.51 \quad = \quad 12\log_2 12 - 3\log_2 3 - 2\log_2 2 - 2\log_2 2 - 1\log_2 1 - 3\log_2 3 - 1\log_2 1.$$

The evaluation for the first plane is 29.51.

The following table shows the evaluations of the other strings in the segregation.

| Object | The string | Alphabet | Frequencies | Evaluation |
|---|---|---|---|---|
| $\text{Plane}_1$ | **cadgbeca** | **abcdg** | 3 2 2 1 3 1 | 29.51 |
| $\text{Plane}_2$ | **okm** | **kmo** | 1 1 1 | 4.25 |
| $\text{Plane}_3$ | **ttrr** | **rt** | 2 2 | 4.00 |
| T | **ABACC** | **ABC** | 2 1 2 | 7.61 |
| Templates | **11312113112** | **123** | 7 2 2 | 14.40 |
| Total | | | | 59.77 |

It might seem intuitively reasonable that the simplest solution would be one in which all phonemes occur on the one plane. But this model in fact requires more information to specify. Here are the evaluations for the monoplane model.

| Object | The string | Alphabet | Frequencies | Evaluation |
|---|---|---|---|---|
| $\text{Plane}_1$ | **catdogbetbreakcream** | **abcdegkmort** | 3 2 2 1 3 1 1 1 1 2 2 | 63.20 |
| T | **AAABB** | **AB** | 3 2 | 4.85 |
| Templates | **11111111** | **1** | 8 | 0.00 |
| Total | | | | 68.06 |

The evaluation of *Templates* and $T$ is lower in this segregation than in the previous one, but this is more than offset by the cost of placing all phonemes on the one plane. As a result, the segregation which appears to be qualitatively more compact, is quantitatively more expensive. The best segregation strikes a balance between the compactness of the templates, which is improved by having fewer planes, and the compactness of the individual planes which is improved by having many planes.

# 3 Results

In order to turn the evaluation measure into a learning system, it must be wrapped in a search. Because of the size of the search space, a non-deterministic weak search was used: *simulated annealing*. The resulting algorithm was implemented in C on VAXen and SUNs. The implementation allows the user to select the maximum number of classes into which the segments can be grouped.

Data from thirty languages was collected in a suitable form for the program. The languages were chosen to fit, as closely as possible subject to the availability of material, the distribution proposed by Bell (1978) to avoid genetic bias. The artificial language Esperanto was also used to see if it behaved differently from natural languages. For each language the data consisted of a word list of at least two hundred and fifty words (preliminary testing suggested that this number was sufficient for convergence) with each phoneme occurring at least five times in the corpus. The

| Language | Classification | Language | Classification |
|---|---|---|---|
| Arabic | C-V | Mandarin | C-CV |
| Auyana | C-V | Martuthunira | (C-C)-V |
| Axininca Campa | C-(C-V) | Miwok | C-V |
| Big Nambas | CV-V | Nez Perce | C-V |
| Daga | C-V | Panyjima | C-V |
| Esperanto | C-V | Piro | C-V |
| Gilbertese | C-V | Siroi | C-V |
| Gothic | CV-V | Swahili | C-V |
| Hungarian | C-V | Telugu | C-V |
| Italian | C-V | Thai | (C-C)-V |
| Ixil | C-V | Tigak | C-V |
| Jacaltec | C-V | Turkish | C-V |
| Japanese | C-V | Wiyot | C-V |
| Karen | (C-C)-V | Wojokeso | C-V |
| Latin | C-V | Yoruba | C-V |

Figure 1: Classifications of phonemes. A class marked C, V, and CV contains non-syllabic, syllabic or both sorts of phonemes respectively. Parentheses indicate the classification when restricted to two classes.

words were taken (with the exception of Gilbertese) from continuous texts, to avoid any possible bias due to citation forms. Arabic data, for example, could be biased if verbs only ever occurred in one binyan. Each word was restricted to one occurrence in the corpus to avoid undue influence of frequently occurring items. Data was taken in phonemic form (using the analyses in the source grammars usually), and the classifications in the source analyses were used for comparison with the program's results.

Tonal information, even though it is phonemic in Yoruba, Mandarin, Thai and Karen, was ignored. This was done because the relationship of tonal markings to a segmental ordering is not clear.

The program was run twice on each data set. In the first run, the number of permitted classes was unrestricted, allowing each phoneme to possibly exist in a class of its own. In all cases, however, the algorithm selected at most three classes, and often only two. The second run restricted the classification to at most two classes.

The results are shown in figure 1.

C indicates a purely non-syllabic class, V a class containing only syllabic items, and CV a class containing both syllabic and non-syllabic segments. The parentheses show how classes are joined when restricted to two classes, if in the unlimited case more than two classes are found. For example, the Axininca Campa phoneme inventory was divided into three groups. Two of these contained only non-syllabic segments (consonants), the third only syllabic ones (vowels). When restricted to two classes, one class of consonants was grouped with the vowels. It is an interesting empirical result that the ternary classifications were always subdivisions of the binary classification.

When restricted to two classes only, the program nearly always divided the phonemes into two classes: one of consonants and the other of vowels. The results were evaluated by comparing them with the classification into consonants and vowels that was given in the source material. This result is gratifying as almost all proposals for planar segregation upto now have segregated phonemes according to these two classes.

There were four exceptions to this result. Even in the exceptions strongly and exclusively syllabic items (such as the vowel *a*) were separated from other strongly and exclusively non-syllabic items (such as *k*). In one case, Big Nambas, there was rampant bivalency: many phonemes acted both syllabically and non-syllabically. This caused problems in the classification due to the restriction that classes be disjoint: the high vowels are classified with the consonants.

In two other cases, Axininca Campa and Gothic, alternate phonemicisations considered viable but not used in the sources, gave better results. Mandarin segregated phonemes which could occur in the rhyme from those which could not. Investigations of Chinese secret languages has uncovered considerable supporting evidence for an onset-rhyme planar segregation (Ellison (ms)).

When a large number of planes were permitted, upto one for each phoneme, the most common result was to retain the two plane syllabic/non-syllabic segregation. Once again there were a few exceptions, all using three planes. In three of these cases, Axininca Campa, Thai and Karen, the split class was divided according to whether the segment occurred exclusively in a particular syllable position: onset, rhyme or coda. In the case of Martuthunira, interestingly, the program separated out the class of consonants which may begin words.

Further investigations are needed to determine whether the program indeed found real linguistic structure in these cases, and, if not, why it failed to arrive at the expected analysis.

# 4 Discussion

The above results show that for a simple but real and non-trivial learning task, a minimalist approach is both possible and successful. The choice of model is constrained only by the theory inherent in the statement of the learning problem and the general Solomonoff learning approach.

Exactly the same algorithm can be used to learn things in a non-phonological domain. For example, let the segment inventory contain digits, metric multipliers (micro-, milli-,centi-,kilo-) and metric measures (litres,metres,grams). If the corpus is a list of measurements (232 kilograms, 19 metres, 2 millilitres, etc.) then the program will quite readily divide the segment inventory into the three classes of digits, multipliers and measures. So the learning algorithm is not domain-specific as would be one confined to determining phonological planar segregation in all and only possible human languages.

It follows therefore that domain-specific knowledge is not a necessity for a system to learn linguistic structure, even when the lack of this knowledge results in large search spaces.

Last but not least, the algorithm provides a new technique for determining planar segregations. It suggests some interesting results for Mandarin and Martuthunira, as well as lending confirmation to the existing analysis for Arabic (compare McCarthy (1981)).

# References


Bell, A (1978) **Language Samples**, in Greenberg (1978).

Ellison, TM (ms) **Onset and rhyme planes in Chinese secret languages.**

Greenberg, JH, ed (1978) *Universals of Human Language*, Stanford University Press, Stanford.

Ladefoged, P (1975) *A Course in Phonetics*, Harcourt Brace Jovanovich, New York.

van Laarhoven, PJM (1987) *Simulated Annealing*, D. Reidel, Dordrecht.

McCarthy, JJ (1981) **A prosodic theory of nonconcatenative morphology**, *Linguistic Inquiry*, 12:373-418.

McCarthy, JJ (1989) **Linear order in phonological representation**, *Linguistic Inquiry*, 20:71-100.

O'Grady, WO (1987) *Principles of Grammar and Learning*, University of Chicago Press, Chicago IL.

Prince, A (1987) **Planes and copying**, *Linguistic Inquiry*, 18:491-509.

Solomonoff, RJ (1964) **A formal theory of inductive inference**, *Information and Control*, 7:1-22,224-254.

# Applying Some CFL Learnability Results to Natural Language Learning

Leona F. Fass

**Abstract:**

An inference process is described, whereby syntactic models for context-free languages (CFLs) may be inductively constructed, and the languages so learned, from suitable linguistic knowledge samples. Properties of learnable generative and recognitive CFL models, and of the knowledge samples requisite for their successful inference, are emphasized. A related testing process is also described, whereby correctness of potential syntactic models is determined by exhaustive experimental means. The adaptation of both processes from the domain of CFL learning to that of natural language learning is next proposed. Then natural languages may be learned (syntactically) through identification in the limit or, otherwise, learned approximately. As time permits, there may be additional discussion of the relationship between these results and such processes as parsing and semantic analysis.

**About the author:**

Leona F. Fass received a B.S. in Mathematics and Science Education from Cornell University and an M.S.E. and Ph.D. in Computer and Information Science from the University of Pennsylvania. Prior to obtaining her Ph.D. she held research, administrative and/or teaching positions at Penn and Temple University. Since then she has been on the faculties of the University of California, Georgetown University, and the Naval Postgraduate School.

Dr. Fass' early research included development of an ALGOL interpreter and additional (CFL) linguistic features for a forerunner of FORTH. She began investigating inductive inference, specifically as a CFL learning technicque, around 1980, and has extended this work into related areas at various times, and from various perspectives, since then. (A representative list of her relevant publications/presentations over the past 10 years appears at the end of this paper.) Her research interests primarily have focused on language structure and processing; knowledge acquisition; and the general interactions of logic, language and computation. She is currently a member of the AAAI, ACL, ACM, ASL, IEEE-CS and LSA.

Partial support for this research was provided by a grant from the NPS Foundation Research Program.

## I. Introduction

Although there are many arguments made against the context-freeness of natural language, most debators agree that much of the natural language L may be syntactically described as a context-free language or, a CFL. Thus techniques applicable to the class of context-free languages may be used to obtain results that (at least) approximately apply to any natural language L. These include results on the definition and effective determination of syntactic models and, in particular, results on their learnability.

Employing what at first appears to be non-traditional CFL sentence representation enables the extension of some traditional machine theory, not previously applicable to the CF language domain. Once this is done, machine learning techniques may be adapted so that generative and recognitive models, precisely characterizing such CFL representations, can be found.

By representing linguistic knowledge suitably, it is shown that unique syntactic models for the knowledge exist, that they are finite, and that they may be determined effectively. There are "complementary" techniques for finding a syntactic model (either a grammar or a recognizer): constructing it by inductive inference from a sample of correct (positive) data; or determining it by exhaustive (positive and negative) data tests. Thus "non-traditional" language representation -- actually closely-related to some traditional CFL parsing techniques -- leads to effective language learning by positive or negative means. Extensions of these results to the case of natural language learning follow, by use of adaptive techniques and approximations.

An overview of main results is presented next, with some attention to related work and possible future research directions.

## II. Representing CFL Knowledge to Define Unique Syntactic Models

Based on a suggestion of Levy and Joshi [19], we represent the sentences of a CFL L not in the usual linear-string fashion, but rather, in a fashion conveying some phrase structure: the skeletons S of their derivation trees (interior labels deleted). Thus we consider, instead of the CFL L, its structured version, S, as defined by some known context-free grammar. This skeletal, tree-like, structured language S is recognizable by a class of bottom-up tree recognizers: the class of skeletal automata that Levy and Joshi first described [19].

By generalizing classical machine theory to the class of structured CFLs, we have shown that each such structured language S has a unique finite-state minimal deterministic recognizer. Corresponding to this unique recognitive characterization is a unique generative characterization: a canonical CF grammar producing the structured language S unambiguously. Relative to similar grammars it, too, is minimal. The components of either of these syntactic models are precisely determined by the structure of the language, conveyed as S. S, generally, is infinite, but is learnable if a characterizing finite syntactic model is acquirable by finite means.

## III. Effective Determination of a Finite Syntactic Model

We have shown that if a language is known to be CF then the constructs of either of the syntactic models described above is inductively inferable from a finite sample of the structures S: specific positive data. [If it is known there is an n-variable backwards-deterministic grammar (i.e., where no two distinct productions have the same right-hand-side) for S, then structures of S of depth

$\leq 2n$ are proven to be a sufficient data sample for inductive inference of a grammar, or skeletal automaton recognizer, characterizing the entire structured language S. If the known grammar is not backwards-deterministic, the sample should be up to depth $2^n$.] Several algorithms for finding the models are described [e.g., 1, 2, 12], including an efficient minimizing algorithm in [7].

Precisely the theory for constructive inference of a correct syntactic model for S is adaptable to the case of testing a potential model of the language, to see whether or not it is (in)correct. Tests on positive and negative data (relative to S) are described, through which a potential model of S is learnable, once, by conclusive testing, it is effectively "verified". [If it is known that depth $\leq 2n$ structures of S define a correct model, through inference, then structures of depth $\leq 2n$ within S and not in S, relative to its defined complement, will sufficiently test a potential model and determine whether or not it is correct.] Thus as long as a language is known to be CF, its syntactic models, as described, are learnable -- through inductive inference or testing -- employing finite positive or negative means.

## IV. Adaptations to Natural Language Learning

If a natural language is known to be context-free, and an n-variable grammar for the language is also known, then all of the above results automatically apply. More likely, though, a sample of language structures will be given, but it really will not be known if the language to-be-learned is (or is not) CF. Here we can show that, if the language really is CF, then the adaptive (monotonic) learning processes we use eventually will discover a correct language model, by identifying it "in the limit" [18]. If, on the other hand, the language is not CF, then our learning algorithms will never halt to provide a model successfully. However, at any point we may cease the inference or testing process and accept that the result we then have is a "learned" model, that "characterizes the language approximately".

## V. Related Results and Future Research Directions

Related results that have come out of this research have included: techniques for "minimizing" CF processors and grammars; generalization of some classical complexity results (with W.I. Gasarch [7]) to the learnable models; and the beginnings of a theory (with E.S. Bainbridge [5], J.C. Cherniavsky, et al.) relating structured-CFL processors to traditional pda processors -- particularly in the case of "easily parsable" languages with LL(k) and LR(k) grammars.

Future research plans include further work in the area of parsing; possible applications of "minimization" results to attribute grammars (semantics), as described in [17]; and adaptations of structural CF knowledge representation to the broader area of natural language acquisition and processing (e.g., as suggested in [20]).

A full paper will expand upon the above concepts and provide illustrative examples, as time and space permit. Many of the results cited in the present paper are proven in the author's work, listed below.

## Selected Relevant Papers and Presentations by the Author

1. L.F. Fass, "Inference of Skeletal Automata", preliminary version (1982), revised as Georgetown University Department of Computer Science Technical Report, TR-02, December 1984.

2. L.F. Fass, "Learning Context-Free Languages from their Structured Sentences", SIGACT News, Vol. 15, No. 3 (1983), pp. 24-35. Abstracted in Zbl. fur Mathematik, Band 528 (Nov. 1984), p. 376.

3. L.F. Fass, "Inference of Context-Free Languages from Structural Descriptions", presented at Center for Study of Language and Information/Association for Symbolic Logic Meeting on Logic, Language and Computation, Stanford, July, 1985. Abstracted in Journal of Symbolic Logic, Vol. 51, No. 3 (Sept. p. 842.

4. L.F. Fass, "Inductive Inference Applied to Context-Free Language Acquisition", presented at The American Philosophical Association/Association for Symbolic Logic Joint Meeting, Washington, D.C., December, 1985. Abstracted in Journal of Symbolic Logic, Vol. 51, No. 4 (Dec. 1986), p. 1090.

5. E.S. Bainbridge and L.F. Fass, "Alternative Structures for Context-Free Languages", preliminary Version (1986). Extended project in progress.

6. L.F. Fass, "On the Inference of Canonical Context-Free Grammars", SIGACT News, Vol. 17, No. 4 (May, 1986), pp. 55-60.

7. L.F. Fass and W.I. Gasarch, "Complexity Issues in Skeletal Automata", preliminary version March, 1987, appears as Computer Science Series, TR2035, University of Maryland, College Park, 1988. Extended version submitted.

8. L.F. Fass, "Knowledge Representation and Inductive Inference of Language", presented at Association for Symbolic Logic/ACM STOC Conjoint Symposium on Computer Science and Logic, New York, May, 1987. Abstracted in Journal of Symbolic Logic, Vol. 53, No. 4 (1988), pp. 1272-1273.

9. L.F. Fass, "Learnability of CFLs: Inferring Syntactic Models from Constituent Structure", presented at 1987 Linguistic Institute, Meeting on the Theoretical Interactions of Linguistics and Logic, Stanford, July 1987. Abstracted in Journal of Symbolic Logic, Vol. 53, No. 4 (1988) pp. 1277-1278. Research Note appears in SIGART Special Issue on Knowledge Acquisition, April 1989, pp. 175-176.

10. L.F. Fass, "On Language Inference, Testing and Parsing", presented at the 1989 Linguistic Institute, Meeting on the Theoretical Interactions of Linguistics and Logic, University of Arizona, Tucson, July 1989.

11. L.F. Fass, "A Common Basis for Inductive Inference and Testing", presented at the Seventh Pacific Northwest Software Quality Conference, Portland, OR, September 1989. Appears in Proceedings, pp. 183-200.

12. L.F. Fass, "A Minimal Deterministic Acceptor for Any (Structured) Context-Free Language", preliminary version (1987). Extended version presented at the 1990-91 Annual Meeting of the Linguistic Society of America, Chicago, January 1991.

13. L.F. Fass, "A Generalization of the Myhill-Nerode Theorem and Some Conjectured Applications" presented at the Association for Symbolic Logic 89-90 Annual Meeting, University of California, Berkeley, January 1990. To be abstracted in the Journal of Symbolic Logic.

14. L.F. Fass, "Learnable, Testable, Finite Models of Language", presented at the 35th Annual Conf. of International Linguistic Association, New York City, March-April 1990.

14. L.F. Fass, "Learnable, Testable, Finite Models of Language", presented at the 35th Annual Conf. of International Linguistic Association, New York City, March-April 1990.

15. L.F. Fass, "Acquiring knowledge by Positive or Negative Means", presented at the Association for Symbolic Logic 90-91 Annual Meeting, Carnegie Mellon, January 1991. To be abstracted in the Journal of Symbolic Logic.

16. L.F. Fass, "Results in Language Learning: Formal and Natural", preliminary version (1991), submitted.

Selected Additional References

17. Aho, A.V., R. Sethi and J.D. Ullman, Compilers: Principles, Techniques and Tools, Addison-Wesley, Reading, MA, 1986.

18. Gold, E.M., "Language Identification in the Limit," Information and Control, Vol. 10 (1967), pp. 447-474.

19. Levy, L.S. and A.K. Joshi, "Skeletal Structural Descriptions", Information and Control, vol. 39 (1978), pp. 192-211.

20. Tomita, M., "LR Parsers for Natural Language", presented at 10th International Conf. on Computational Linguistics (COLING 84), Stanford 1984. In Proceedings, pp. 354-357.

The author may be contacted at mailing address:

Dr. Leona F. Fass
P.O. Box 2914
Carmel, CA 93921

* * * * *

Janet Dean Fodor
Linguistics Department, Graduate Center, City University of New York.

## RESEARCH INTERESTS

I have worked on theoretical syntax and semantics, and several aspects of psycholinguistics. I know no computer science, but try to follow developments in computational linguistics. I am particularly interested in sentence processing, and favor an interdisciplinary problem-oriented approach which focusses on a particular question and draws on all available methodologies to answer it. The annual CUNY Conference on Human Sentence Processing was founded to foster this approach and provide a forum for sharing research results and expertise among linguists, psychologists, and computer scientists.

In recent years I have been following the development of non-transformational theories of syntax, particularly GPSG and HPSG, and have been working with Stephen Crain (Linguistics Dept., U. of Connecticut) to evaluate these theories against psycholinguistic data. We have sketched how sentence processing would proceed if based on a GPSG grammar, and have argued that this model accounts for recent experimental results at least as well as a transformational (Government Binding theory) model. Debate on these issues is summarized in my papers in Language and Cognitive Processes 4, SI 155-209, 1989; and in T. Wasow, P. Sells and S. Shieber (eds.) Foundational Issues in Natural Language Processing, MIT Press, 1991.

As reflected in my paper for this symposium, Crain and I have also been working on a model of language acquisition based on GPSG/HPSG theory. 'Poverty of the stimulus' arguments present a serious challenge to any acquisition model, even assuming considerable innate knowledge. The principles-and-parameters approach of Government Binding theory offers one solution to these problems. Crain and I set out to elucidate why no other kind of solution could be successful, but convinced ourselves instead that phrase structure theory does provide a basis for a simple, non-reflective acquisition algorithm. We would be happy to hear from anyone interested in implementing a learning model of this kind.

## RELEVANT PUBLICATIONS

Fodor, J. D. and S. Crain (1987) Simplicity and generality of rules in language acquisition. In B. MacWhinney (ed.) Mechanisms of Language Acquisition, Lawrence Erlbaum Associates.

Fodor, J.D. (1989) Learning the periphery. In Learnability and Linguistic Theory, R. Matthews and W. Demopoulos (eds.) Kluwer Academic Publishers. (Expanded version of paper by same title in CUNYForum 1986.)

Fodor, J. D. (1989) Principle-based learning. CUNYForum 14, 59-67.

Fodor, J. D. (1990) Parameters and parameter-setting in a phrase structure grammar. In L. Frazier and J. de Villiers (eds.) Language Processing and Language Acquisition, Kluwer Academic Publishers, Dordrecht.

Fodor, J. D. and Crain, S. (1990) Phrase structure parameters. Linguistics and Philosophy 13.6, 619-659.

Fodor, J. D. (in press) Learnability of phrase structure grammars. In R. Levine (ed.) Formal Linguistics: Theory and Implementation, Vancouver Studies in Cognitive Science; University of British Columbia Press.

Fodor, J. D. (in press) Islands, learnability and the lexicon. In H. Goodluck and M. Rochemont (eds.) Island Constraints: Theory, Acquisition and Processing, Kluwer Academic Press, Dordrecht.

## MAKING PHRASE STRUCTURE GRAMMARS LEARNABLE

### ABSTRACT

GPSG and HPSG (Generalized Phrase Structure Grammar, Head-driven Phrase Structure Grammar) are theories of language structure not of behavior, but we can ask whether the grammars they define could be learned under psychologically natural conditions. In fact they cannot. Their language-specific rules could be learned only by a cumbersome and unreliable hypothesis-formation-and-testing device. Their language-specific constraints cannot be acquired at all if learners receive no systematic negative input. And the grammars hypothesized by a GPSG/HPSG learner cannot be guaranteed to respect the Subset Principle: an incomplete grammar for the target language typically generates a superset of the target, and without negative data there is no motive for the learner to move to the more complex but more restricted target grammar.

I argue that learnability can be achieved, without loss of descriptive adequacy, by five revisions of current GPSG/HPSG. These prevent overgeneration by lexical metarules and linear precedence statements, and most importantly they replace language-specific constraints with universal defaults that can be overridden by acquirable rules. I call the resulting system LPSG (Learnable Phrase Structure Grammar). It turns out that these revisions which make learning possible in principle also greatly simplify the learning process. No hypothesis-formation-and-testing procedures are necessary. The learner need only strip off predictable feature specifications from the input, by applying innate feature instantiation principles in reverse. Rule learning in LPSG is thus simple and 'mechanical'; and unlike a parameter-setting device, it works uniformly for both the core and the periphery of language.

## MAKING PHRASE STRUCTURE GRAMMARS LEARNABLE

GPSG and HPSG (Generalized Phrase Structure Grammar, Head-driven Phrase Structure Grammar) have been presented as theories of language structure, not of behavior. (See Gazdar et al. 1985; Pollard and Sag 1987, in press.) But it is of interest nevertheless to ask whether the grammars they define could be used for sentence processing, and whether they could be learned, under psychologically natural conditions. With regard to learnability, GPSG/HPSG grammars do not fare well. There are three main reasons for this, essentially identical to the problems that afflicted the learning of transformational grammars before the advent of modern parameter-setting models.

(1) GPSG/HPSG grammars contain language-specific constraints, which cannot be acquired on the standard assumption that language learners receive no systematic negative input (= information about what is not a sentence of the language). A grammar which lacks a needed constraint will generate a proper superset of the target language. Thus it will accommodate every positive datum the learner will encounter, so s/he will have no motive to add the constraint.

(2) GPSG/HPSG grammars contain language-specific rules, couched in syntactic feature notation. Again the problem is that an incomplete grammar for the target language will generate a superset of the language. If a learner omits a feature specification from a rule, the result will be a broader rule which overgenerates. Without negative data the learner could not recognize the necessity of adding the feature specification.

(3) The fact that GPSG/HPSG grammars constitute a mix of rules and constraints creates descriptive ambiguities. A learner faced with a novel datum would not know whether to add a new rule, or relax an existing constraint, or some combination of the two. Thus learning cannot be deterministic. Rather, it appears to require some kind of hypothesis-formation-and-testing (HFT) device, which can experiment with the alternatives and select between them on the basis of further data. But HFT devices are complex, and cumbersome in operation. Either they engage in a vast trial-and-error search through the space of possible grammars, or their convergence on the correct grammar is difficult to guarantee.

In short: GPSG/HPSG grammars are unlearnable in principle because there is no way to guarantee that the interim grammars hypothesized by learners will obey the Subset Principle (Berwick 1985), which requires learners without negative data to start with the most conservative grammar and to proceeed to more powerful ones only when that is necessary to accommodate further (positive) data. And even if learning were possible in principle, the available learning procedures appear to compare very poorly in practice with the sort of 'mechanical' triggering of parameter switches that suffices for learning Government Binding theory grammars.

It can be shown, however, that this learnability failure is not an inherent property of phrase structure grammars, or of rule-based systems in general. Everything depends on how the rules interact with other components of the grammar. In particular, I argue that learnability can be achieved, without loss of descriptive adequacy, by the following five revisions of current GPSG/HPSG:

(i) No language-specific FCRs (Feature Co-occurrence Restrictions) or FSDs (Feature Specification Defaults).

(ii) The Specific Defaults Principle: a specific (i.e., non-disjunctive) default value must be assigned by Universal Grammar to every feature in every context, unless the value in that context is universally fixed or is universally free.

(iii) The Double M Convention: if a rule contains two or more optional marked feature specifications, only one marked value may be selected for the same local tree, unless the rule explicitly indicates that they may co-occur.

(iv) Linear Precedence statements must characterize permitted orders of sister constituents, not required orders.

(v) Lexical (meta)rules do not preserve subcategorization features. Subcategorization features are category-valued, not integer-valued. (Already so in HPSG.)

I refer to the resulting system as LPSG (Learnable Phrase Structure Grammar). Though I have no proof that (i) - (v) are jointly sufficient for learnability, I know (at present) of no other modifications that are needed.

Amendments (iv) and (v) block the overgeneralizing tendencies of linear precedence rules and lexical rules; I will not discuss these further here. Amendments (i) and (ii) are the most central. Their joint effect is to translate language-specific constraints (FCRs) in GPSG/HPSG grammars into universal default statements (FSDs) in LPSG. The result, as I will illustrate below, is a system in which rules differ in markedness, depending on how many of their feature values need to be explicitly specified and how many can be omitted because they follow from the universal defaults or from other general principles. Revision (iii) then exploits this markedness system to block the generalization of rules from unmarked to marked values, which is dangerous, while permitting safer generalizations from marked to unmarked values.

How does LPSG's greater reliance on default feature specifications make learning possible? Because all of its defaults are universal, we can assume they are innate, so the problem of learning them without negative data does not arise. The defaults, together with any absolute universal constraints, will constitute a learner's initial hypothesis about the target language, prior to any experience. Since the defaults in LPSG embody all possible language-specific constraints, the learner's initial hypothesis will be maximally restricted, as the Subset Principle requires. Learning will consist of progressively loosening these restrictions, where necessary, by adding language-specific rules to override the defaults. It is thus the rules, not the constraints, that capture variation between languages. Note that rules, unlike constraints, can be learned from positive data. And since each rule adds to the complexity of the grammar, we can assume that learners won't adopt a rule until or unless the data require it, so learning will be conservative. Finally, though LPSG rules are expressed in feature notation, the omission of a feature no longer licenses indiscriminate generation of trees with either (any) value of that feature. Instead, only the default (unmarked) value is licensed. If that value matches the input, the rule can remain underspecified; if the input has the marked value instead, the mismatch will force the learner to complicate the rule by specifying the marked value.

Consider the learning of language-specific patterns of extraction by WH-movement (or the phrase structure analogue of such movement, using the feature SLASH). Extraction is very limited in Slavic languages, it is less restricted in English, and it is freer still in Scandinavian languages. A somewhat over-tidy summary of the relevant language facts is shown in Figure 1 (for more linguistic details see Cichocki 1983, Engdahl 1982).

Figure 1. Extraction facts (simplified) assumed here:

| | | Polish | English | Swedish |
|---|---|---|---|---|
| Extraction from matrix VP | (Who do you like?) | yes | yes | yes |
| Extraction from object compl. | (Who does John think that you like?) | no | yes | yes |
| Extraction from WH-compl. | (Who does John know whether you like?) | no | no | yes |

Note that with respect to these extraction facts, Polish is a proper subset of English, which is a proper subset of Swedish. The Subset Principle therefore requires that it is the English and Swedish learners who must do the learning; the strict constraints on Polish must be innately established as the initial hypothesis. But in GPSG, where the differences between the three languages are captured by language-specific constraints, the relative complexity of their grammars predicts exactly the opposite of this. As sketched in Figure 2, GPSG predicts that Polish learners have more learning to do, more constraints to acquire, than English or Swedish learners.

Figure 2. Language-specific constraints in GPSG:

| | | Polish | English | Swedish |
|---|---|---|---|---|
| Constraint: | no extraction over WH | + | + | |
| Constraint: | no extraction across S | + | | |

In LPSG, by contrast, these constraints will have the status of universal defaults, innate, not needing to be learned. What must be learned is rules to override these defaults, to permit extraction where it does occur. And as Figure 3 shows, in terms of rules the relative complexity of the three grammars is in keeping with the Subset Principle.

Figure 3. Universal default and language-specific rules in LPSG:

| | Polish | English | Swedish |
|---|---|---|---|
| Rule: can extract over WH | | | + |
| Rule: can extract across S | | + | + |
| Default: no extraction over S or WH | + | + | + |

The feature-omission problem is illustrated in Figure 4. In GPSG, a simple rule posited by the learner to account for a non-extraction structure will generate not only that structure but also the corresponding extraction structure.

Figure 4. Free instantiation of SLASH in GPSG:

```
Input:    VP          Motivates rule:  VP  -->  H , S[FIN]     Rule also licenses:    VP[SLASH NP]
         /   \                                                                       /    \
        V     S[FIN]                                                              V      S[FIN, SLASH NP]


Input:    VP          Motivates rule:  VP  -->  H , S[WH]      Rule also licenses:    VP[SLASH NP]
         /   \                                                                       /    \
        V     S[WH]                                                               V      S[WH, SLASH NP]
```

The feature SLASH carries information between a WH-phrase and its trace, licensing the 'extraction'. GPSG permits free instantiation of SLASH, subject only to universal constraints (such as the Head Feature Convention); that is, GPSG construes non-specification of SLASH in a rule as licensing local trees both with and without a SLASH feature. So once again, Polish learners would learn Swedish by mistake. A Polish learner would encounter the non-extraction constructions, and would thereby have acquired the corresponding extraction constructions. As shown in Figure 5, all three languages would have the same rules, which would overgenerate in Polish and English.

Figure 5. Overgenerating rules in GPSG:

| | | Polish | English | Swedish |
|---|---|---|---|---|
| Rule: VP --> H , S[FIN] | (non-extraction, extraction) | + | + | + |
| Rule: VP --> H , S[WH] | (non-extraction, extraction) | + | + | + |

The cure for this in LPSG is the Specific Defaults Principle (= revision (ii) above). This requires every feature (or: every feature whose value needs to be learned in some natural language) to have a default value, which will be supplied in a tree whenever the value is left unspecified in a rule. Let us assume, as seems reasonable, that the default for SLASH is for it to be absent, to have no value (in other words: non-extraction is the unmarked case). Then the rules in Figure 5 will generate only non-extraction constructions without SLASH. Different rules, more highly specified ones containing explicit SLASH features, will be necessary to generate the extraction constructions, as shown in Figure 6. A learner of Polish will not encounter the trees that would motivate these more elaborate rules with SLASH, and therefore will not overgenerate the extraction constructions.

Figure 6. Same rules subject to default in LPSG, don't overgenerate:

| | | Polish | English | Swedish |
|---|---|---|---|---|
| Default: -SLASH (unless +NULL) | | + | + | + |
| Rule: VP --> H , S[FIN] | (non-extraction) | + | + | + |
| Rule: VP --> H , S[WH] | (non-extraction) | + | + | + |
| Rule: VP[SLASH NP] --> H , S[FIN, SLASH NP] | (extraction) | | + | + |
| Rule: VP[SLASH NP] --> H , S[WH, SLASH NP] | (extraction) | | | + |

The grammars for English and Swedish look more complex in LPSG than in GPSG. But (a) the relative complexities are now right for learnability; (b) the SLASH rules can be collapsed with the basic rules into more general rule schemata, so the extra complexity is in fact very slight; (c) in compensation, LPSG grammars lack the language-specific constraints of GPSG.

Most interestingly, it turns out that these revisions which make learning possible in principle also greatly simplify the learning process. An LPSG learner has no constraints to acquire, but only (lexical entries and) rules.

So there is no descriptive ambiguity as in standard GPSG/HPSG. When a learner encounters a novel local tree, one not licensed by his current grammar, his only choice is to add a rule (or add a feature option to an existing rule schema, which is equivalent to adding a rule and then collapsing it into the schema). Furthermore, since phrase structure rules are merely schematic characterizations of legal local trees, acquiring a new rule does not call for any creativity, any reflection, or hypothesis-formation-and-testing procedures, but is a simple routine matter. The worst that could happen is that the learner simply adopts the novel tree as a new rule in the grammar just as it stands. But that of course would lead to an unnecessarily vast and redundant grammar. To achieve an optimal grammar, the learner needs to strip off from the novel local tree all feature specifications that are predictable on the basis of universal principles and defaults. And to do this, all he need do is apply the principles and defaults (with which he is innately equipped) in reverse to the novel local tree. The feature specifications that remain after this feature stripping process will constitute the schematic rule his grammar needs; it will be the minimal characterization of just what is idiosyncratic to that syntactic construction in that language.

Thus rule learning in LPSG is simple and 'mechanical', and does compare well with a parameter-setting model though completely different from it in its details. LPSG also makes it possible for phrase structure learning to satisfy various other desiderata for an optimal learning device. For example:

where I = a novel input which initiates a learning event;
$G_i$ = learner's grammar at the time that I is encountered;
$G_{i+1}$ = the grammar the learner adopts in response to I;
L(G) = the grammar licensed (generated) by grammar G:

(i) $G_{i+1} = G_i$ if $G_i$ licenses I. [Prevents unnecessary grammar changes.]

(ii) $G_{i+1}$ licenses I. [Prevents fruitless grammar changes.]

(iii) $G_{i+1}$ is as small as possible consistent with (ii) and (iv). [Simplicity metric; permits reductions in grammar size ('restructuring'), but no unnecessary increases.]

(iv) $L(G_{i+1})$ includes as many sentences of $L(G_i)$ as possible, compatible with (ii). [Prevents loss of prior learning, but allows retreat from errors - tho' not from Subset Principle violations.]

These conditions (or others similar to them) can help direct grammar choice in profitable directions, and greatly reduce the amount of random trial and error before convergence on the correct grammar, thus bringing the learning model closer to a psychologically plausible account of actual language learning. Whether such conditions can be implemented without unrealistically complex computations depends on how the learning device operates. For example, Wexler and Culicover (1980) imposed condition (ii) on addition of a transformational rule to the grammar, but could not impose it on deletion of a transformation because it was too difficult to identify a suitable rule to delete. But all the conditions above, as well as the Subset Principle, are easily implemented in LPSG; in fact they fall out quite naturally from the feature stripping mechanism.

Finally, there is one respect in which feature-stripping is arguably superior to GB's parameter-setting. GB avoids the familiar drawbacks of hypothesis-testing by assuming that designated inputs automatically trigger the re-setting of a parameter. But the price for this is that the parameters, their values, and their triggers must all be innately listed; hence they must be finite, and for a plausible model they should be relatively few in number. This is why parameter-setting has been proposed only for 'core grammar'; a completely different (hypothesis-testing?) learning device is needed in addition for acquiring the more varied and unpredictable 'periphery' of a natural language. But LPSG needs no such duplication of learning mechanisms. Its default principles define a single continuum of markedness covering core and periphery alike. Rules are more costly to specify the more peripheral they are, i.e., the more they depart from the universal defaults. But the same feature-stripping learning device will acquire them all.

There are some matters needing further attention which I will not be able to address here. For example: the feature-stripping device is essentially cost-free, since it utilises feature instantiation principles which must in any case be innately provided and used in constructing sentence derivations. But it does need to be established that this derivational algorithm can apply efficiently in reverse as well as in the forward direction (i.e., to vacuum predictable feature values off trees, rather than to spray them on). Also, to the extent that the collapsing of rules into schemata is essential for achieving streamlined adult grammars, it must be ascertained that the rule collapsing

process is information preserving. With the increase in number of default statements in LPSG, it may be necessary to establish priority principles (e.g., an 'elsewhere condition') to determine which ones should override which others in case of conflict. The process by which learners parse novel input needs to be explored. Since the current grammar fails, by definition, to license a novel sentence type, the learner must apparently guess how to structure it. We need to know to what extent this guessing is linguistically guided. Finally, it needs to be shown that the LPSG learning mechanism is (or can be made) resilient to misleading or ungrammatical input. Though I have some thoughts on each of these points, it seems to me that by far the most practical way to investigate them further is by computer implementation of the feature-stripping process, and I would be grateful for any advice or assistance that this audience can offer.

References


Berwick R. C. (1985) The Acquisition of Syntactic Knowledge. Cambridge, MA: MIT Press.

Cicocki, W. (1983) Multiple WH-questions in Polish: A two-comp analysis. In Toronto Working Papers in Linguistics 4, 53-71.

Engdahl, E. (1982) Restrictions on unbounded dependencies in Swedish. In E. Engdahl and E. Ejerhed (eds.) Readings on Unbounded Dependencies in Scandinavian Languages (Umea Studies in the Humanities 43). Stockholm, Sweden: Almqvist & Wiksell International.

Gazdar, G., Klein, E., Pullum, G. K. and Sag, I. A. (1985) Generalized Phrase Structure Grammar. Cambridge, MA: Harvard University Press.

Pollard, C. and Sag, I. A. (1987) Information-Based Syntax and Semantics, Volume 1: Fundamentals (CSLI Lecture Notes Number 13). Stanford, CA: CSLI.

Pollard, C. and Sag, I. A. (in press) Information-Based Syntax and Semantics, Volume 2: Binding and Control (CSLI Lecture Notes). Stanford, CA: CSLI.

Wexler, K. and Culicover, P. W. (1980) Formal Principles of Language Acquisition. Cambridge, MA: MIT Press.

# A Case-Based, Inductive Architecture for Natural Language Processing

Marc Goodman

Computer Science Department
Center for Complex Systems
Brandeis University
415 South Street
Waltham, MA 02254-9110

Cognitive Systems, Inc.
234 Church Street
New Haven, CT 06510

## Research Interests

I have been involved with the construction of commercial Natural Language systems since January of 1985. Between 1985 and 1987, my primary interest was the construction of a General Lexicon of English, including the syntactic and semantic knowledge necessary to apply this lexicon to a variety of domains. In 1987 I began leading a research effort into Case-Based Reasoning which lead to a generic CBR shell. This shell has been applied, or is being applied to, problems in battle planning, network fault diagnosis and recovery, credit worthiness evaluation, credit collection, geological classification, and machine tool fault diagnosis and recovery. In 1988, I began work on applying this shell to message classification, and created the PRISM message classifier which was presented at IAAI-90. In addition to my work with Parse-O-Matic, I am currently involved in applying CBR/Adaptive Planning to spatial reasoning and representation, with particular emphasis on issues of Natural Language ties to spatial reasoning.

## Abstract

Recent work with Case-Based Reasoning in the areas of Battle Projection [Goodman, 1989] and Telex Classification [Goodman, 1990] indicate that this approach holds the potential for building and fielding large, knowledge-based systems which are faster, more accurate, and require significantly less time to knowledge-engineer and maintain than with other approaches. Additionally, CBR provides a memory-based framework for knowledge representation which simplifies interaction between sources of knowledge, facilitates the handling of generalizations and exceptions, and supports learning from success and failure [Kolodner and Riesbeck, 1990].

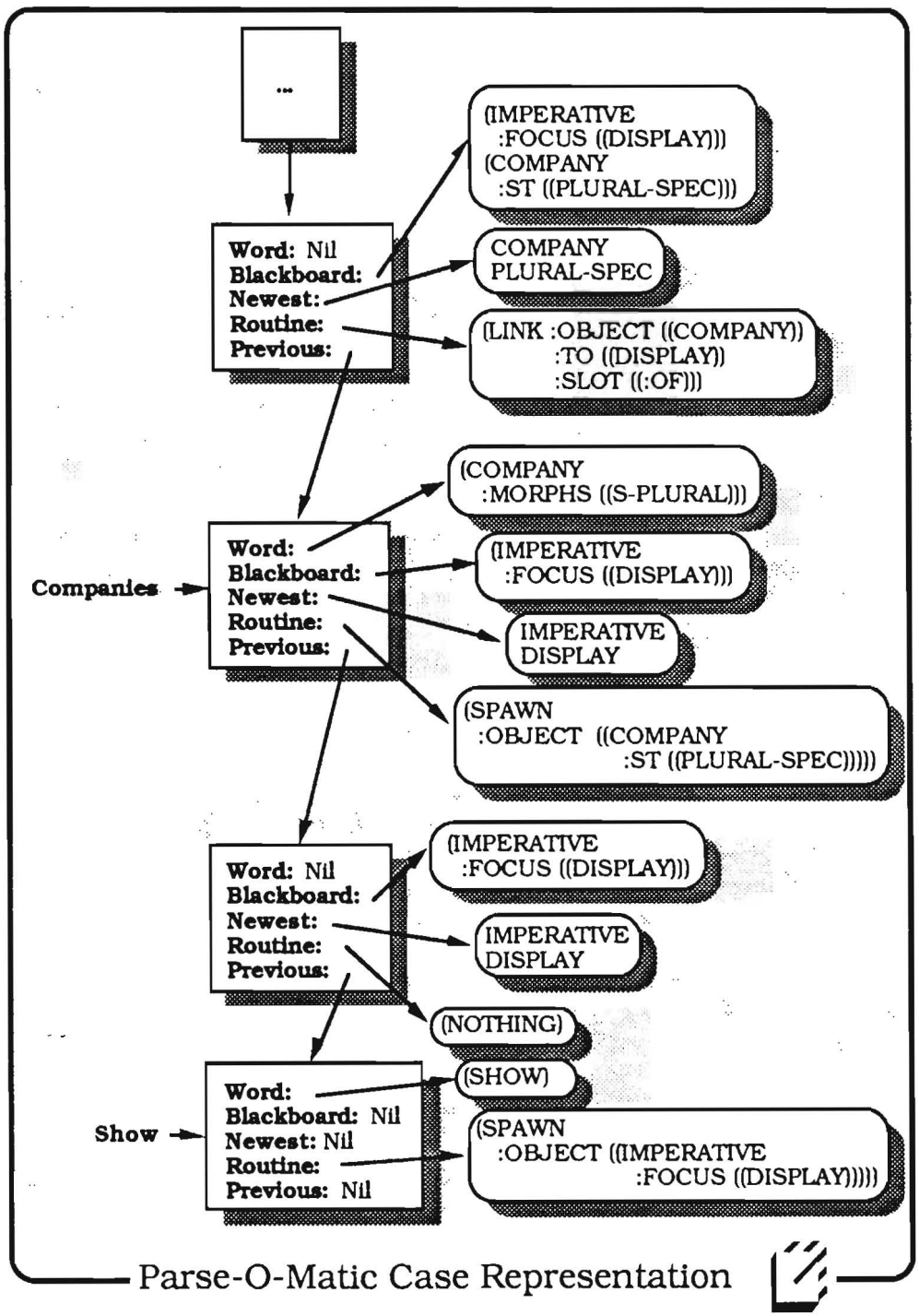
Parse-O-Matic, a system which builds frame-based semantic representations of Natural Language Requests, exploits these characteristics of CBR. Comparative knowledge engineering time and accuracy are given for Parse-O-Matic, and the KNET parser [Strong, 1989].

---

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by the Air Force Office of Scientific Research under Contract No. F49620-88-C-0058. The United States Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon.

## Case Representation

Parse-O-Matic represents knowledge of parsing based on a Dynamic Theory of Activity [Agre and Chapman, 1987]. Each case contains a single step (or *routine*) describing a representation-building action to take in a particular context. The derivation of a representation for a particular request may consist of several routines. Since the system has decomposed derivations, retrieval has the effect of dynamically combining individual steps of old derivations to create new derivations.

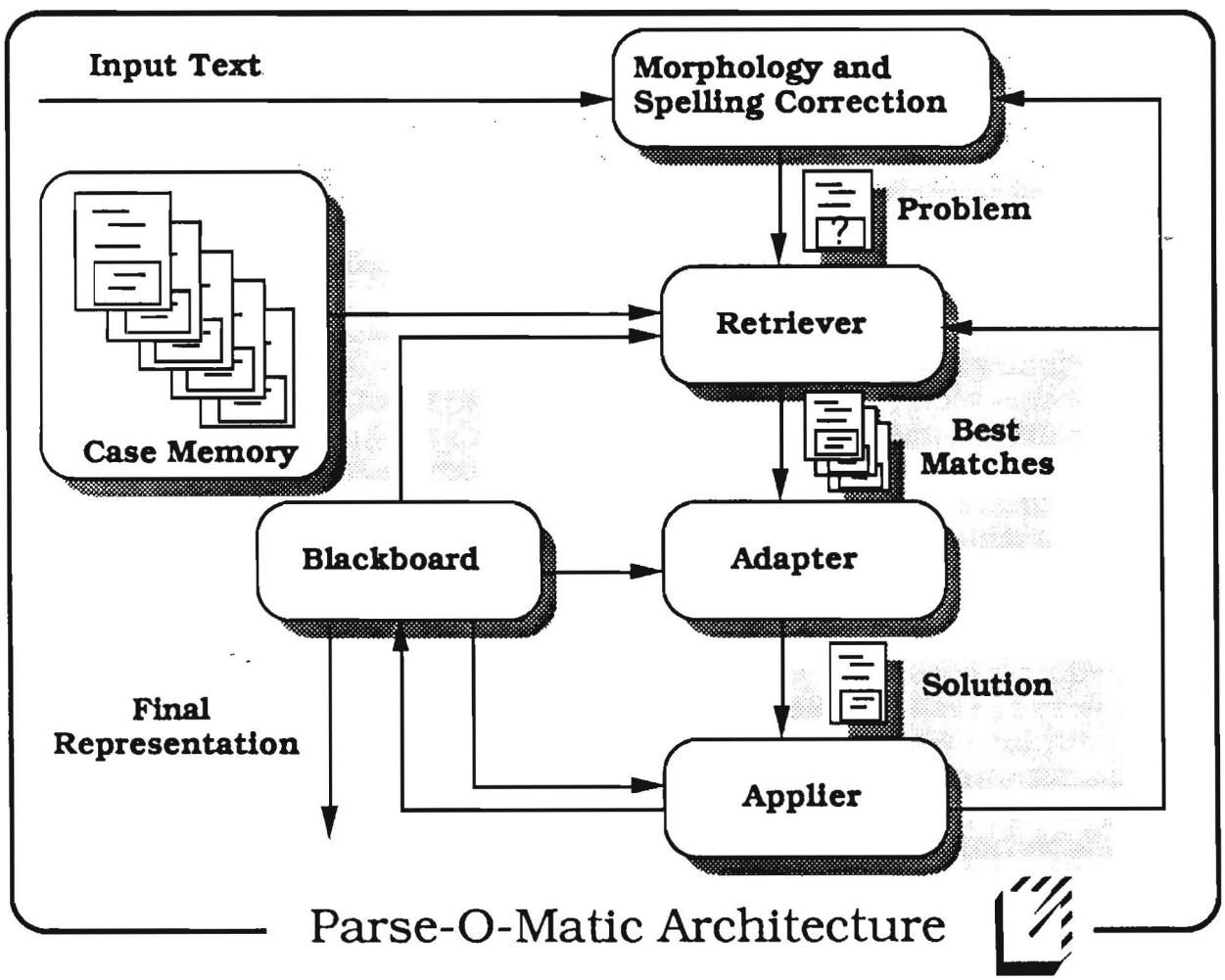


In the construction of Parse-O-Matic, we based our representational choice on a previous parser (the KNET Parser [Strong, 1989]) and representational scheme. The KNET Parser used a forward-chaining, blackboard-based production system, compiled into a RETE net, with an underlying Truth Maintenance System to build a Frame-Based Semantic representation. The action side of each production allowed the manipulation of the representation on the blackboard, including actions such as spawning new frames onto the blackboard, linking frames to other frames (representing a role-filling relationship), changing the type of frames, removing frames, unlinking frames from other frames, etc. Parse-O-Matic, therefore, represents routines as frames describing the manipulation of frames on a blackboard.

In addition to a particular routine, each case representation must also include a set of features which can be used for case retrieval. These features are used to determine what makes cases similar with respect to their routines, and serve as the basis for index generation. An example of the complete case representation is the sentence "Show companies..." with later cases at the top and earlier cases at the bottom:

**Parse-O-Matic Architecture**

This case representation leads to the following architecture for Parse-O-Matic:



Parsing proceeds as follows: an input text is passed into a module which performs word-by-word morphological analysis and spelling correction. The first word is then passed to a case retriever, which uses the word, the current representation on the blackboard, and the previous cases (if any) to create a new case. This case is then used to traverse a set of indices, and a set of best-matching cases is returned from the case memory. The routine indexed on the retrieved cases is then adapted, using information on the blackboard, to create a new routine. This new routine (the solution) is passed into an applier which may fetch frames from the blackboard, and post frames to the blackboard or modify frames on the blackboard. The applier then passes control back to the retriever (if an action was performed), or passes control back to morphology and spelling correction (if no additional action is required in the current context). When control is passed back to the retriever, a new case is created with the current blackboard representation and the previous case, and the process repeats. When control is passed back to morphology and spelling correction, the next word is passed to the retriever and the process repeats. If no words remain, Parse-O-Matic is finished with the text and the final representation is on the blackboard.

**Case Indexing and Retrieval**

Case indexing in Parse-O-Matic is based on an extended version of the Automatic Interaction Detection algorithm [Hartigan, 1975]. The technique applies an analysis of variance model in order to partition a sample into a series of non-overlapping subgroups whose means explain more of the variance in outcome than any other set of subgroups. In operation, it is very similar to the CART algorithm [Brieman, Friedman, Olshen and Stone 1984].

Parse-O-Matic uses three techniques for reducing the total number of features considered in indexing cases. The first technique is to only consider a small subset of cases while generating each discrimination. One technique for reducing the number of cases required is to preselect cases based on certain attributes, such as a good mix of outcomes, before generating the discrimination.

A second technique is to break indexing into several passes, each of which considers a heuristic subset of the total features which are frequently meaningful. Two examples of this technique are: 1) pre-index on the last frames spawned into working memory since subsequent links, removals, etc. will usually refer to these new frames, and 2) pre-index on features which are necessary preconditions to executing the routine, such as the presence or absence of conceptual types referenced by the routine. In all, Parse-O-Matic uses 10 separate indexing passes based on different limiting heuristics.

The third technique for reducing the total number of operations is to incrementally add new cases into an existing library and to generate only those indices which are required to account for variations in the new cases. Using this technique, significant development work can be done without fully reindexing the library. The disadvantage of this approach is that cases which are incrementally added are only used to generate splits near the leaves of the index tree, whereas they could be useful in generating appropriate discriminations earlier in indexing.

Since case retrieval consists of traversing a binary discrimination tree of indices, which is O(log(n)) on the number of cases in the library, and the total number of

retrievals is roughly linear on the number of words, m, O(m log(n)) parsing is possible with this architecture.

**Case Adaptation**

Much of Parse-O-Matic's power comes from its ability to adapt previous routines to new situations. A simple example of such an adaptation is a sentence like "Show companies with beta under 5." Let's say that during a certain point in its derivation of a representation for this sentence, the best-matching case has a routine like:

```
(LINK :OBJECT ((COMPANY))
      :TO ((BOOK-VALUE-OF))
      :SLOT ((:OF)))
```

while our current representation looks like:

```
(IMPERATIVE-1
    :FOCUS ((DISPLAY-1
                :OBJECT ((COMPANY-1 :ST ((PLURAL-SPEC-1)))))))
(BETA-OF-1 :IS ((NUMBER-1)))
```

Since there is no BOOK-VALUE-OF frame in the representation, Parse-O-Matic must adapt this routine to fit the current situation. Parse-O-Matic does so using Local Search [Kolodner and Riesbeck, 1990]. The missing role-filler, BOOK-VALUE-OF, is generalized to ATTRIBUTIVE-RELATIONSHIP using the conceptual hierarchy. The blackboard is then searched for a frame which inherits from ATTRIBUTIVE-RELATIONSHIP. In this case, it finds BETA-OF-1. The routine is then reinstantiated with the new role fillers, yielding:

```
(LINK :OBJECT ((COMPANY))
      :TO ((BETA-OF))
      :SLOT ((:OF)))
```

A more difficult adaptation, which Parse-O-Matic does not currently support, is a case such as "Show companies with earnings per share over 6." Let's assume that we've spawned an EARNINGS-OF relationship on the word earnings, and on "share" we get back the nearest routine:

```
(CHANGE :OBJECT ((BOOK-VALUE-OF))
        :TO ((BOOK-VALUE-PER-SHARE-OF)))
```

while our current representation looks like:

```
(IMPERATIVE-1
    :FOCUS ((DISPLAY-1
                :OBJECT ((COMPANY-1 :ST ((PLURAL-SPEC-1)))))))
(EARNINGS-OF-1 :IS ((MONEY-AMOUNT-1))
               :OF ((COMPANY-1)))
```

Changing the :OBJECT role-filler of the routine to EARNINGS-OF-1 from BOOK-VALUE OF is the same as above. However, we would like Parse-O-Matic to adapt the routine as follows: 1) BOOK-VALUE-PER-SHARE-OF is the :PER-SHARE type attribute of BOOK-VALUE-OF, 2) EARNINGS-OF-1 is being substituted for BOOK-VALUE-OF, 3) Query Memory to find the :PER-SHARE type attribute of EARNINGS-OF (which would be EPS

OF), 4) reinstantiate the routine with EPS-OF substituted for BOOK-VALUE-PER-SHAI OF yielding:

```
(CHANGE :OBJECT ((EARNINGS-OF))
        :TO ((EPS-OF)))
```

Plans are under way for extending Parse-O-Matic to deal with this kind of adaptation. Parse-O-Matic currently requires a brute-force approach (i.e. adding examples for all such CHANGEs to case memory) to get this.

**Conclusion**

Parse-O-Matic views Natural Language Processing as a memory-intensive process. Its Case-Based architecture allows episodic knowledge to be added in a localized, incremental fashion. Generalizations and exceptions over lexical, semantic, and syntactic constructions are handled automatically through Inductive indexing of the case library.

Parse-O-Matic and the KNET parser have both been applied to the same Natural Language domain where Parse-O-Matic achieved a comparable accuracy to the KNET parser (over 90% accuracy) in roughly 50% of the knowledge engineering time. Parse-O-Matic also parses more quickly (in about 25% of the time taken by the KNET parser).

**References**

[Agre and Chapman, 1987] P. Agre and D. Chapman. "Pengi: An Implementation of a Theory of Activity." In Proceedings of the Sixth National Conference on Artificial Intelligence, Seattle, WA, 1987, pp. 268-272.

[Brieman, Friedman, Olshen and Stone 1984] L. Brieman, J. Friedman, R. Olshen, and C. Stone. "Classification and Regression Trees." Wadsworth, Belmont, CA, 1984.

[Goodman, 1989] M. Goodman. "CBR in Battle Planning." In Second Proceedings of a Workshop on Case-Based Reasoning, Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1988, pp. 312-326.

[Goodman, 1990] M. Goodman. "Prism: A Case-Based Telex Classifier." In Proceedings of the Second Conference on Innovative Applications of Artificial Intelligence, American Association for Artificial Intelligence, Menlo Park, CA, 1990, pp. 86-90.

[Hartigan, 1975] J. Hartigan. "Clustering Algorithms." John Wiley & Sons, 1975.

[Kolodner and Riesbeck, 1990] J. Kolodner and C. Riesbeck. "Case-Based Reasoning: Introduction, Overview, and Algorithms." Georgia Institute of Technology, Department of Information and Computer Science, Atlanta, GA, and Northwestern University, Institute for the Learning Sciences, Evanston, IL, 1990.

[Strong, 1989] R. Strong. The KNET Parser Documentation Set. Cognitive Systems, Inc., New Haven, CT, 1989.

## THE SYMBOL GROUNDING PROBLEM AND CATEGORICAL PERCEPTION

Stevan Harnad, Department of Psychology, Princeton University, Princeton NJ 08544

**Research Interests (S. Harnad):** My current research interest is in the symbol grounding problem and categorical perception: A symbol system is a set of physical tokens (e.g., scratches on paper, holes on a tape, flip-flop states in a computer) and rules for manipulating them (e.g., erase "0" and write "1"). The rules are purely syntactic: They operate only on the (arbitrary) shapes of the symbols, not their meanings. The symbols and symbol combinations can be given a systematic semantic interpretation, for example, they can be interpreted as meaning objects ("cat," "mat") or states of affairs ("the cat is on the mat"). The meanings of the symbols, however, are not grounded in the symbol system itself; they derive from the mind of the interpreter. Hence, on pain of infinite regress, the mind cannot itself be just a symbol system, syntactically manipulating symbols purely on the basis of their shapes. This is the "symbol grounding problem."

How can one ground the meanings of symbols within the symbol system itself? This is impossible in a pure symbol system, but in a hybrid system, one based bottom-up on nonsymbolic robotic functions such as transduction, analog transformations and sensory invariance extraction, the meanings of elementary symbols can be grounded in the system's capacity to discriminate and categorize (name) the external objects and states of affairs that its symbols refer to, based on the projections of those objects and states of affairs on its sensory surfaces. The grounded elementary symbols ("cat," "mat") can then be rulefully combined and recombined to form higher-order symbols and symbol strings ("the cat is on the mat") that inherit the grounding as nonarbitrary constraints on their shapes.

Harnad, S. (ed.) (1987) *Categorical Perception: The Groundwork of Cognition*. New York: Cambridge University Press.

Harnad, S. (1989) Minds, Machines and Searle. *Journal of Theoretical and Experimental Artificial Intelligence* 1: 5-25.

Harnad, S. (1990) Against Computational Hermeneutics. Invited commentary on: Eric Dietrich's "Computationalism," *Social Epistemology* 4: 167-172.

Harnad, S. (1990) Lost in the Hermeneutic Hall of Mirrors. Invited Commentary on: Michael Dyer: "Minds, Machines, Searle and Harnad," *J. of Exp. Theor. A.I. 1990* 2: 321 - 327.

Harnad, S. (1990) Symbols and Nets: Cooperation vs. Competition. Review of: S. Pinker and J. Mehler (Eds.) (1988) "Connections and Symbols" *Connection Science* 2: 257-260.

Harnad, S. (1990) The Symbol Grounding Problem. *Physica D* 42: 335-346.

## Categorical Perception and the Evolution of Supervised Learning in Neural Nets

Stevan Harnad*, Stephen J. Hanson*,** Joseph Lubin*,
*Princeton University, **Siemens Research Center

**Abstract:** *Some of the features of animal and human categorical perception (CP) for color, pitch and speech are exhibited by neural net simulations of CP with one-dimensional inputs: When a backprop net is trained to discriminate and then categorize a set of stimuli, the second task is accomplished by "warping" the similarity space (compressing within-category distances and expanding between-category distances). This natural side-effect also occurs in humans and animals. Such CP categories, consisting of named, bounded regions of similarity space, may be the ground level out of which higher-order categories are constructed; nets are one possible candidate for the mechanism that learns the sensorimotor invariants that connect arbitrary names (elementary symbols?) to the nonarbitrary shapes of objects. This paper examines how and why such compression/expansion effects occur in neural nets.*

**Categorical Perception and the Evolution of Supervised Learning in Neural Nets**

S Harnad*, SJ Hanson*,** & J Lubin*
*Princeton Univ., **Siemens Res. Ctr.

## 1. Categorical Perception

One of the most remarkable properties of human perception is that it seems to carve the world at its joints. The physical signals that bombard our sensory surfaces do not give rise to a "blooming, buzzing confusion" but to relatively orderly experiences, segmented into "chunks" (Miller 1956) or categories. How does our brain sort things into categories on the basis of the sensory signals it receives?

A relevant phenomenon in human and animal perception that has received a good deal of attention is "categorical perception" (CP) (Harnad 1987): Equal-sized physical differences in the physical signals arriving at our sensory receptors are perceived as smaller within categories and larger between categories. For example, differences in wavelength within the range we call "yellow" are perceived as smaller than equal-sized differences that straddle the boundary between yellow and the range we call "green." The wavelength continuum has somehow been "warped," with some regions getting compressed and other regions getting stretched out.

In the case of color CP, although learning may have played a role, most of the warping seems to have been done by evolution, with the result that it is probably an inborn property of our sensory systems, modifiable only minimally (if at all) by experience. Other prominent examples of CP have been found in human speech perception as well as in some animal signalling systems (see chapters in Harnad 1987 for examples). These too seem to be largely innate, although they are modifiable by experience. Musical pitch categories may be examples of CP effects that arise primarily as a result of learning. CP effects have also been reported to occur purely as a result of learning in experiments with artificial continua; similar "warping" effects might be expected to arise from learning complex multidimensional categories, as in learning to sort baby chicks as male and female, or histological slides as cancerous or noncancerous.

The generation of CP (enhanced within-category similarity and enhanced between-category differences) by perceptual learning has been described as the "acquired similarity [difference] of cues" but no mechanism has been proposed to explain how or why it occurs.[1]

In this paper we will show how CP might arise as a natural side-effect of the means by which certain standard neural net models (backpropagation, Rumelhart & McClelland 1986) accomplish learning. They acquire the capacity to sort their inputs into the categories imposed by supervised learning through altering the pairwise distances between them (where distance is the degree to which a pair of inputs is discriminable by the net) until there is sufficient within-category compression and between-category separation to accomplish reliable categorization. As we shall see, however, the nets don't necessarily stop at a minimal degree of compression/separation; rather, they overshoot, producing much stronger CP effects than seem necessary to accomplish the categorization.

CP is of interest not only in its own right, as a very basic perceptual phenomenon, but also as a possible contributor to solving the "symbol grounding problem" (Harnad 1990): In a formal symbol system such as a computer program, or in the actual implementation of such a system on a machine, symbols are manipulated on the basis of formal rules or algorithms that apply to the *shapes* of the symbols, not their meanings (i.e., symbol manipulation is syntactic rather than semantic). The meanings of the symbols

[1] Behaviorists proposed an associative explanation -- that members of the same category grew more similar because they were were more closely associated with one another and with their shared category name than with members of different categories and their names, but this is more a restatement of the phenomenon than a model that explains it. The" motor theory of speech perception" explained speech CP by the similarities and differences between the motor pattern required to produce, say, a BA and a DA, but this model applies only to the special case of speech, where there is a perception/production analogue, and has given rise to decades of unfruitful debate about whether or not speech is "special." The last "theory" of CP is the Whorf Hypothesis, according to which CP is a manifestation of how language and culture shape our view of reality. This too seems more a restatement of the phenomenon than an explanation of it.

are projected onto them by the user who interprets the symbols and the symbol manipulations; they are not intrinsic to the system itself. By contrast, if, using the sensory projections on its transducer surfaces, a robot were able to discriminate and categorize the real-world objects, events and states of affairs to which its symbols can be interpreted as referring, then those symbols would be grounded in the robot's causal capacity rather than just being parasitic on the meanings an interpreter projects onto them.

So there is a close connection between the sensorimotor capacity to carve the world at its joints and the cognitive capacity to produce symbolic descriptions of that world: For the compressed and separated "chunks" of the similarity space originating from our sensory receptors can be given names, and those category names can then be combined syntactically to form propositions about the world. Whatever mechanism successfully maps the sensory projections onto their category names is also what grounds the symbol system.

It is one possible candidate mechanism for mapping simple sensory inputs onto category names that will be analyzed here, and in particular, the dynamical role that the warping of similarity space which is characteristic of CP may be playing in its successful performance.

**2. Learning to Split a Line.**

Both the neural net architecture and the task used were very simple. A backpropagation net with 8 input units, 2 - 12 hidden units and 8 or 9 output units was used. The net's task was to learn to sort 8 "lines" into 2 categories (let us call them "short" and "long"). The lines were represented in 6 different ways, in order to test the effects of the input coding. One variable of interest was the "iconicity" of the coding (i.e., how analog, nonarbitrary, or structure-preserving it was in relation to what it represented).

The lines were either "place" coded (e.g., a line of length 4 would be 0 0 0 1 0 0 0 0) or "thermometer" coded (e.g., line 4 would be 1 1 1 1 0 0 0 0). The place code was assumed to be more arbitrary and the thermometer code more analog, in that the thermometer code preserved some multi-unit constraints whereas the place code did not. In addition, the thermometer-coded lines and the place-coded lines could be discrete-coded (as above) or they could be coarse-coded, allowing some gaussian spillover to adjacent units (e.g., line 4 coarse/place-coded might be 0 .001 .1 .99 .1 .001 0 0, and line 4 coarse/thermometer-coded might be .90 .99 .99 .90 .1 .001 0 0). Finally, because CP concerns the formation of boundaries between categories, a lateral inhibition coding was also tested, in which adjacent coarse-coded units were inhibited so as to enhance boundaries (e.g., line 4 lateral-inhibition/place-coded might be .1 .1 .001 .99 .001 .1 .1 .1, and line 4 lateral-inhibition/thermometer-coded would be .8 .9 .9 .99 .001 .1 .1 binary coding, again because it preserved multi-unit constraints. Lateral Inhibition was likewise more analog than the discrete code, but also more complicated, because the width and placement of the boundary effects from the lateral inhibition could in principle help or hinder the formation of a CP boundary, depending on whether the two effects happened to be in or out of phase.

In human experiments the CP effect is defined as an interaction between discrimination (the capacity to tell pairs of stimuli apart, a relative judgment) and identification (the capacity to categorize or name individual stimuli, an absolute judgment). Normally, along a one-dimensional stimulus intensity continuum the discrimination function is log-linear (i.e., equal-sized logarithmic increases in stimulus intensity produce equal-sized increases in sensation intensity, and hence response measures of it, such as same/difference and degree of similarity judgments). CP is a systematic departure from this log-linearity, with relative compression (attenuation) of discriminability within categories and/or relative dilation of discriminability (separation) between categories. The neural net accordingly had to be given an initial discrimination function, which could then be re-examined after categorization training to see whether it had "warped."

The method used to generate the precategorization discrimination function was "auto-association" (Hanson & Kegl 1987; Cottrell, Munro & Zipser 1987). Different nets were trained, separately for each of the 6 representations of the 8 lines, to produce as output exactly the same pattern they received as input. For

each net trained to a predefined criterion level of performance on auto-association the interstimulus distances for all pairs of the 8 lines were then calculated as the euclidean distance between the vectors of hidden unit activations for each pair of lines. For example, if there were four hidden units and their activation values after training for line X were (x1 x2 x3 x4) and for line Y (y1 y2 y3 y4), then the distance between the two inputs, and hence their discriminability for that net, would be the distance between X and Y (see Hanson & Burr 1990 for prior work on using this internal measure of interstimulus distance).

After auto-association the trained weights for the connections between the hidden layer and the output layer were reloaded (and then all weights were left free to vary) and the net was given a double task: Auto-association (again) and categorization, i.e., lines 1 - 4 had to be given one (arbitrary) "name" and lines 5 - 8 had to be given another (e.g., "short" and "long"). In practice, this naming required one more bit on the output, the usual eight for the auto-association, and then one more for the categorization (initially seeded randomly with weights in the (-1.0, 1.0) range).

For each of the six representations, 50 auto-association nets were trained, and the results of each of these were used to train 10 categorization nets; except where noted, the results reported here refer to averages. Once each net was trained on the categorization task, the pairwise interstimulus distances were again computed, as before, and then compared to their precategorization values for that net. A CP effect was defined as a decrease in within-category interstimulus distances and/or an increase in between-category interstimulus distances relative to the auto-association-alone baseline.

**3. Results.**

We will first report the results for auto-association alone, and then for the pre/post comparison. Finally, we will analyze some of the details of the evolution of the CP effects that were observed.

The auto-association-alone results for each of the 6 representations for 4-hidden-unit nets are shown in the corresponding upper portions of Figure 1a-f. Plotted are the interstimulus distances (computed as described earlier) between each pair of inputs for the trained net. As expected, the most arbitrary representation (discrete/place) produced the flattest discrimination function: All interstimulus distances were equal. To an extent, this is true of all the place-coded representations, but it can be seen that the effect of the coarse coding produces some rounding and spillover. All the thermometer-coded representations are more iconic (in the sense that a monotonic increasing relationship, sometimes even a linear one is maintained as the pairs move further apart on the continuum, as in human discrimination functions). This seems to be reflected equally by the discrete/thermometer and coarse/thermometer codes, but the coarse/thermometer code has some more of the properties of human discrimination, as we will see later. The lateral inhibition representations are more complicated, because of interactions between the (arbitrarily chosen) size of the lateral inhibition envelope and the interstimulus increment.

The lower portions of Figure 1a-f show the difference between the interstimulus distances for auto-association alone and the interstimulus distances for auto-association-plus-categorization for each of the six representations. A positive deviation means that the interstimulus distance has decreased and a negative deviation after categorization means it has increased.[2] Hence positive deviations within categories (compression) and/or negative deviations between categories (separation) would be CP effects. As is clear from Figure 1, pronounced CP effects occurred for all 6 representations. (Although there may be some trend toward greater magnitude CP effects with the more iconic representations, the scales vary and the relative magnitude is probably not comparable across representations with this methodology.)

Having observed strong CP effects in all representations, our next question was: Why were they there and what, if anything, were they

[2] To facilitate comparison, the 28 possible pairwise comparisons of the 8 lines are displayed in terms of the size of the increment: Lines differing by 1 unit first, then 2 units, etc. Note that because the category boundary was between lines 4 and 5, increments of 4 or greater are all between-category differences.

for? To examine this more closely we first hypothesized that CP effects may arise as a consequence of compressing the input data into a smaller number of hidden units, so we re-ran the nets with hidden units varying in number from 2 - 12, predicting that the CP effect would diminish with more units. We also thought that whereas a small number of hidden units may give rise to global representations, a large number would allow local ones to form. The prediction was that the global representations would show more of a CP effect.

The categorization task turned out to be very difficult to learn with only 2 hidden units; most nets did not succeed even after a very large number of training trials. With 3 there was CP just as there had been with the 4-hidden-unit nets in Figure 1, and CP continued to be present even when the number of hidden units was increased to 12, exceeding the number of input units. So CP is not merely a consequence of compression. With more hidden units, however, there was more overall separation and less compression in all directions superimposed on the CP effect, both within and between categories.

The next hypothesis was that CP might arise gradually after the first point of separation in the task, as the net overlearned to more extreme values. However, when we trained nets just to the first epsilon of separation and checked for CP, we found the CP pattern was already there then, smaller than in Figure 1, but present.

Another test was whether CP might be an artifact of using the same net, with reloaded weights, to do the auto-association as well as the auto-association-plus-categorization. Now, in some respects this seems the natural thing to do: After all, *we* are the same systems that do discrimination as well as categorization. So although it was a bit like comparing apples and oranges (or at least like making between-subject rather than within-subject comparisons, we also compared performance averaged over many nets for auto-association alone with performance averaged over many other, independent nets, for auto-association-plus-categorization. Here too, although the effect was much weaker and not present in all representations, there was still evidence of a CP effect.

A final test concerned iconicity and interpolation: Was the CP restricted to trained stimuli, or would it "spill over" (or "generalize") to untrained ones? Nets were trained on auto-association the usual way, and then, during categorization training, some of the lines were left untrained (say, line 3 and line 6) to see whether they would nevertheless "warp" in the "right" direction. We found interpolation of the CP effects to untrained lines, but only for the coarse-coded representations.

Our provisional conclusion was that, whatever was responsible for it, CP had to be something very basic to how these nets learned, in particular, to how they accomplished supervised category learning. So the next step was to look more closely at the time-course and evolution of the learning itself. Instead of looking only at the pre/post-categorization comparison of the inter-stimulus distances, we analyzed how the inter-stimulus distances evolved across trials for each of the 8 stimuli. For this we used nets with 3 hidden units. This gave us a visualizable 3-dimensional hidden unit space in which we could follow the locus of the representation of each of the lines in hidden unit space during the course of learning. The results are shown in Figure 2.

Three factors were found to influence the generation of the CP during the course of learning. Two were related to the sigmoid or logistic activation function and one was related to the degree of iconicity of the input representation.

First, a finite, bounded hidden unit space arises because the units saturate to 0 and 1. In the three-dimensional case illustrated here, the hidden unit representations for each of the inputs move into the farthest corners of the unit cube during the course of auto-association learning, maximizing their pairwise distances from one another. This extreme cornering was found with the discrete/place coding (Fig. 2a); there was movement into corners and edges with the discrete/thermometer coding. The other representations showed less of this tendency to move to the extreme periphery of hidden unit space.

This separation tendency thus interacts with the second factor, the iconicity of the thermometer-coded and coarse-coded inputs: Some hidden

unit representations are forced by the auto-association to stay closer to one another than they would otherwise have "liked" to stay because of the input structure they are constrained to inherit (see Figure 2b). Thermometer-coded and coarse-coded inputs accordingly arrive at the categorization stage after auto-association with linearly separable[3] configurations of hidden-units representations whereas place-coded inputs may arrive with more random configurations (depending on the random initial "seeding" values given to each of the weights prior to learning) and hence more of them may fail to be linearly separable (hence failing to be categorizable) after categorization training. Thermometer- and coarse-coded inputs produce faster and more reliable CP effects than place-coded inputs, in that they rarely or never get caught in the local minima that may block linear separability (cf. Figs. 2c - 2e).

The third factor is peculiar to categorization learning and arises from the dynamics of the learning (again because of the logistic function): Because of the error metric of the learning equation, the hidden-unit representations will be pushed with a force that is inversely proportional to an exponential function of their distances from the (hyper)plane separating the two categories.

The codings that generated the largest number of nets that were unable to learn the categorization task were the 2 most arbitrary (noniconic) ones, discrete/place (Fig. 2e) and especially lateral-inhibition/place. Our diagnosis is that with place-coding the output of the auto-associator is more likely to generate configurations in hidden-unit 3-space in which the representations of the eight lines are not readily linearly separable into the two 4-member categories imposed by the task. More training trials are hence required to move such nets into a configuration where the the eight representations are linearly separable (see Figure 2d). The lateral inhibition probably acts to add bumps to the representational space and hence to the error surface. Sometimes the configuration even gets trapped in a local minimum, in which case the categorization cannot be learned at all (see Figure 2e).

So what can so far be inferred about the evolution of CP learning can be stated as follows: During auto-association the iconic properties of the inputs are "imprinted" onto them, and are then reflected in their interstimulus distances in hidden-unit space. Apart from having to remain faithful to these constraints, the effect of auto-association is to maximize the pairwise interstimulus distance among all the stimuli within a bounded, finite space. The categorization phase then has no choice, if it is to generate successful performance, but to "warp" the finite space of this maximal separation, moving some of the stimuli (those within the same category) closer together than they would "like" in order to successfully separate them from the others (those in the other category); the magnitude of the warping effect is proportional to the distance of each stimulus from the plane that marks the boundary between the two categories. A complicating factor, and one affecting either the magnitude of the CP or the probability or number of trials before successful performance is attained, is the initial structure of the 8 stimuli at the end of successful auto-association and the beginning of categorization training: If their initial configuration is at odds with the partition that is needed, more warping is needed, and in some particularly bad configurations (arising mostly with lateral-inhibition-place coding) convergence may not be possible at all.

**4. Conclusions.**

We have analyzed how one particular family of neural nets accomplishes categorization by "warping" interstimulus similarity space in a way that resembles human categorical perception. Other kinds of nets generate CP too (e.g., unsupervised ones), but this analysis seems to be especially revealing about supervised learning, an important form of learning, because the contingencies of survival and successful behavioral adaptation do not always follow the natural lay of the land: Or, to put it another way, where nature's joints are may not be at all obvious from the input alone. Supervision in the form of feedback from the consequences of *mis*categorization may be our best guide as to how to carve up objects, events and states of

[3] Two sets of points in a plane are "linearly separable" if and only if they can be divided into their respective categories by a straight line cutting across the plane. In three dimensional space, linear separability is accomplished by a plane; in higher dimensions, by a hyperplane, etc.

affairs. If so, then the plasticity afforded by a mechanism that can "warp" the landscape in the service of the partition dictated by behavioral contingencies would be a useful one indeed, especially when the behavior is symbolic, and the task is not just to survive, reproduce and get around in the environment, but to describe and explain it -- a mechanism that allows you to "see" the world differently as you carve out ever subtler categories with the fine edge of human language.

**References**

Cottrell, Munro & Zipser (1987) Image compression by back propagation: an example of extensional programming. *ICS Report 8702, Institute for Cognitive Science, UCSD.*

Hanson & Burr (1990) What connectionist models learn: Learning and Representation in connectionist networks. *Behavioral and Brain Sciences 13:*

Hanson, S. J. and Kegl, J. (1987) Parsnip: A Connectionist Model that Learns Natural Language Grammar from Exposure to Natural Language Sentences. *Ninth Annual Cognitive Science Conference, Seattle.*

Harnad, S. (ed.) (1987) *Categorical Perception: The Groundwork of Cognition*. New York: Cambridge University Press.

Harnad, S. (1990) The Symbol Grounding Problem. *Physica D* 42: 335-346.

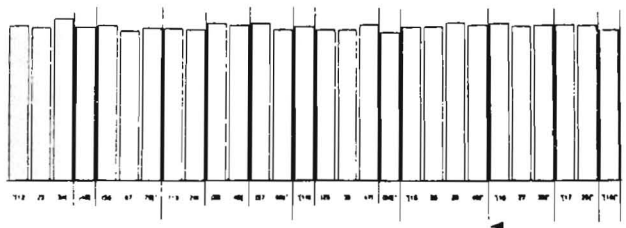
McClelland, J.L., Rumelhart, D. E., and the PDP Research Group (1986) *Parallel distributed processing: Explorations in the* microstructure of cognition," Volume 1. Cambridge MA: MIT/Bradford.

Miller, G. A. (1956) The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review* 63: 81 - 97.

**Figure 1.** Pairwise distances between the 8 lines in hidden-unit space (4 hidden units) for each of the 6 input representations: discrete/place (1a), coarse/place (1b), lateral-inhibition/place (1c), discrete/thermometer (1d), coarse/thermometer (1e), and lateral-inhibition/thermometer. In each case the upper figure displays the pairwise distances following auto-association alone and the lower figure displays the difference between auto-association alone and auto-association plus categorization. The polarity of these differences is positive if the interstimulus distance has become smaller (compression) and negative if it has become larger (separation). To visualize within-category and between-category effects more easily, the comparisons have all been ordered as follows: first the one-unit comparisons 1-2, 2-3,... 7-8; then the two-unit comparisons 1-3, 2-4, etc, and so on until the last seven-unit comparison: 7-8. Note that the category boundary is between stimuli 4 and 5, hence all pairs that cross that boundary are between-category comparisons; otherwise they are within-category comparisons. Almost without exception, within-category distances are compressed and between-category distances are expanded by the categorization learning. Notice also that interstimulus distances before categorization (auto-association alone) tend to be equal (flat) for the more arbitrary codes (discrete/place, lateral-inhibition/place) and ascending with increasing distance in units for the more iconic representations (thermometer and coarse codes).

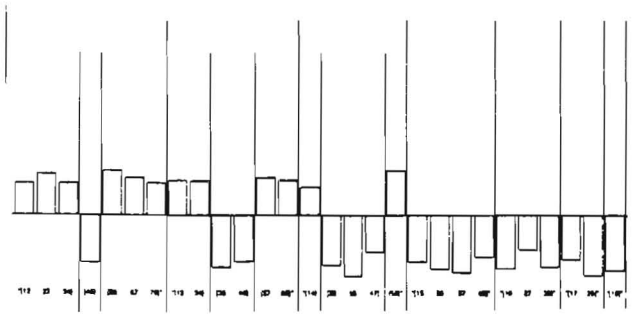
**Figure 2.** The evolution of the 8 line representations in hidden-unit space for 3-hidden-unit nets. Each line's representation is displayed as a point in the unit cube, its value on each axis corresponding to the activations of each of the hidden units (the connecting lines are just to help visualize in 3 dimensions). Figure 2a shows how the arbitrary discrete/place codings evolve during auto-association from their initial random configuration (left) to extreme separation in the corners and edges of the space after auto-association learning (right). Figure 2b, again auto-association alone, shows how the iconic factors in the coarse/thermometer representation constrain this separation. Figure 2c shows the evolution of categorization with the iconic discrete/thermometer code from the final configuration after auto-association alone (left) to the configuration after successful category learning (right). Figure 2d shows in four stages from left to right the more difficult evolution of the configuration with the arbitrary discrete/place code; after considerable movement, linear separability between the two categories is achieved. Finally, Figure 2e shows a discrete/place net that cannot accomplish categorization because it is stuck in a local minimum in which the two categories are not linearly separable.

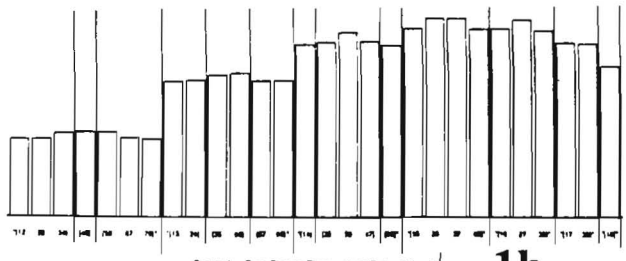
# Figure 1



Series 1 -- Discrete Place -- Auto-Categ (Mean)

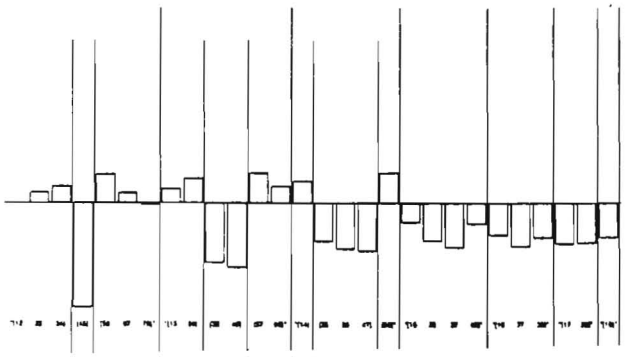
1a





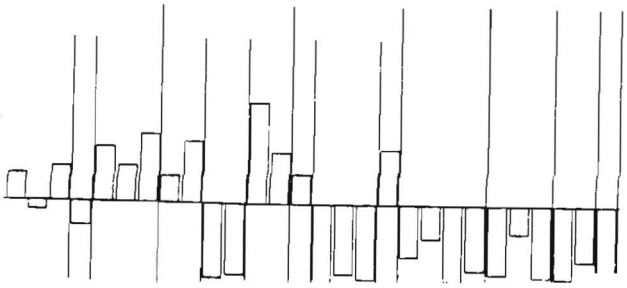
Series 1 -- Prop Coarse Place -- Auto-Categ (Mean)

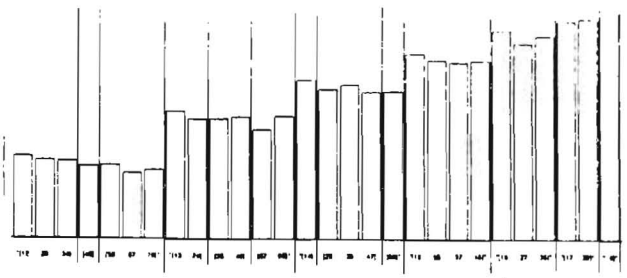
1b



Series 1 -- Lat Inhib Place -- Auto-Categ (Mean)

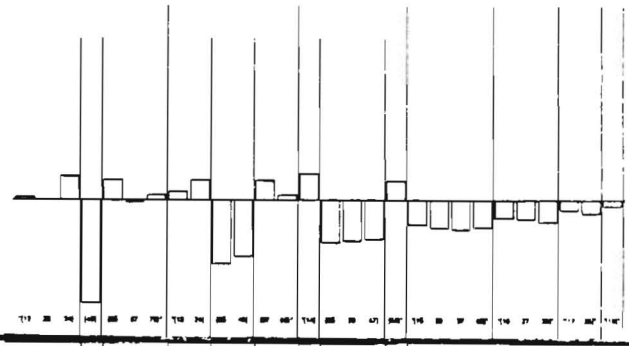
1c

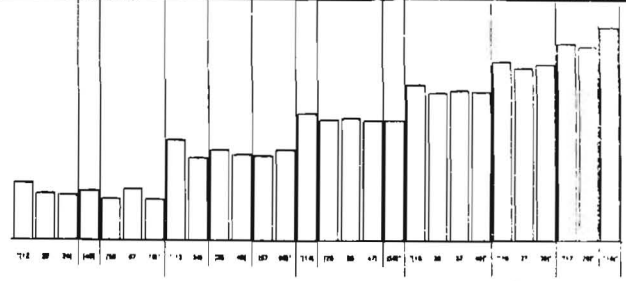




Series 1 -- Discrete Line -- Auto-Categ (Mean)

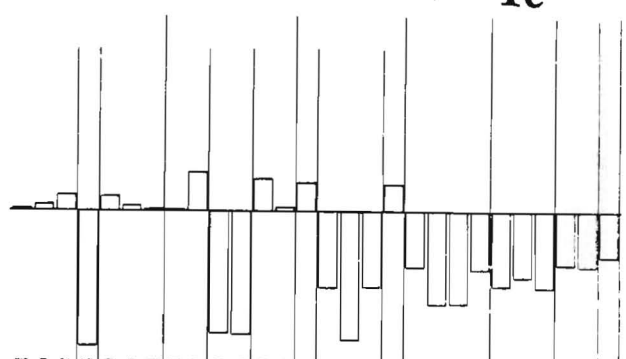
1d

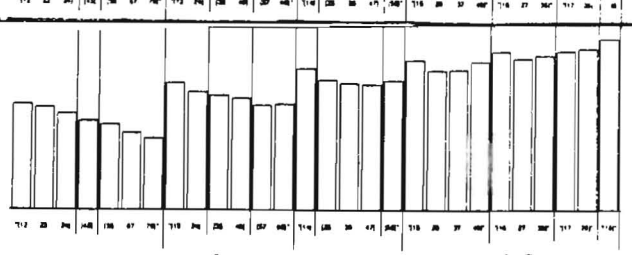




Series 1 -- Prop Coarse Line -- Auto-Categ (Mean)

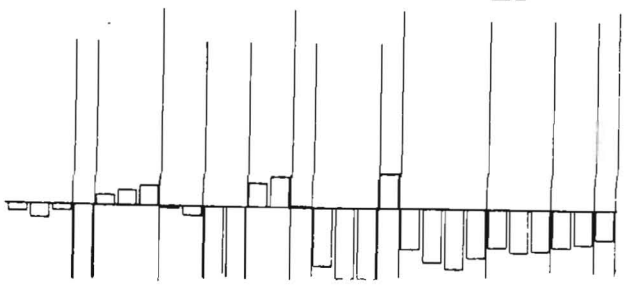
1e



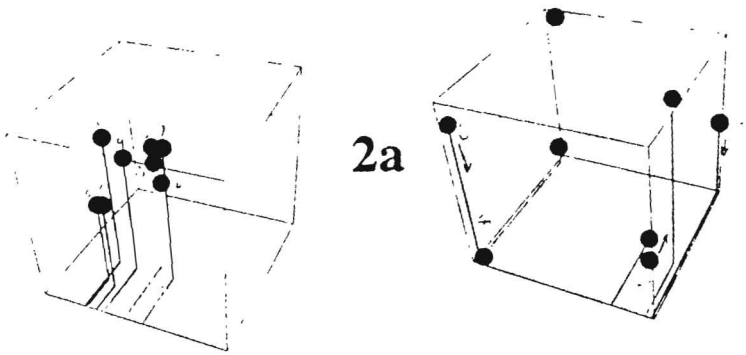


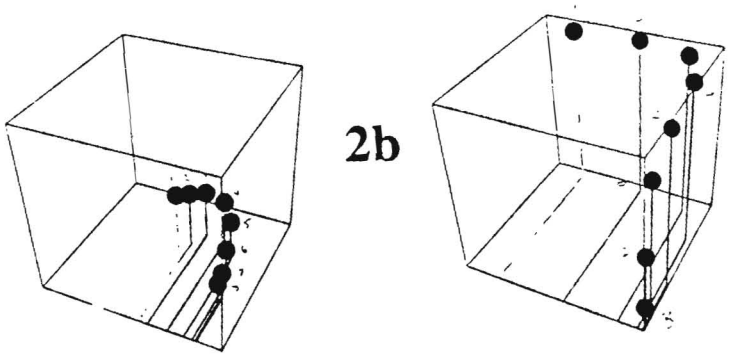
Series 1 -- Lat Inhib Line -- Auto-Categ (Mean)

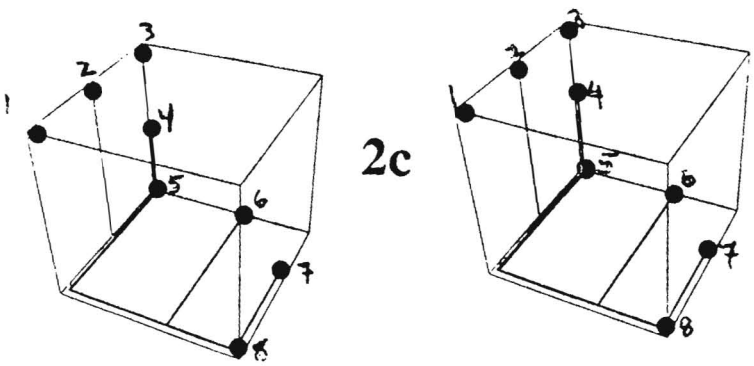
1f

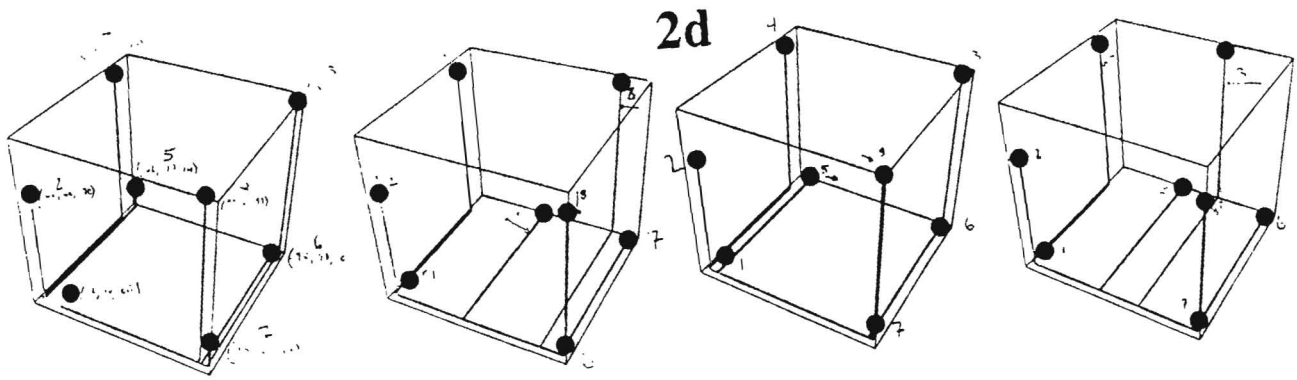


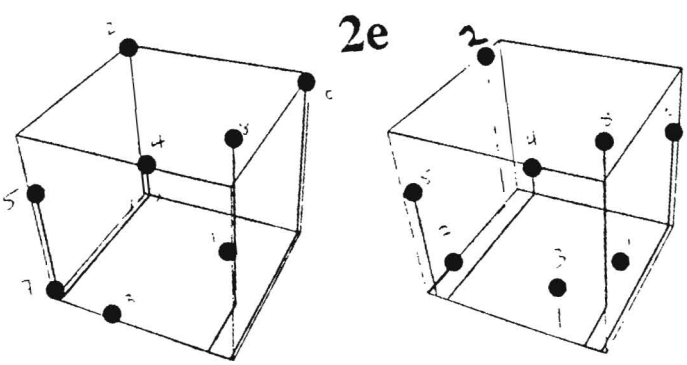
# Figure 2



# Recurrent Neural Networks for Natural Language Acquisition

William Hart
University of California, San Diego
Computer Science and Engineering 0114
9500 Gilman Dr.
La Jolla, CA 92093-0114
whart@cs.ucsd.edu}

My current research with Peregrine Systems involves the development of a natural language interface for information retrieval applications. We treat this project as a machine learning task and are applying neural network technology to the problem of language acquisition. In the following, I will outline three major issues our research is addressing and then mention other elements of my background which are relevant to this workshop.

The first issue involves the nature of the learning system's environment as well as the role of the learning system in that environment. Because we consider a system which acquires natural language through learning, interactions between the learner and its environment play a prominent role in the system's development. Environmental factors can often determine how well the system can learn, as well as what exactly is learned. For example, we are exploring the effect of using phonological vs. normal word representations on learning rate and accuracy in simple learning tasks.

Another important environmental concern regards the method of reinforcement which is applied to the learner. Should it be direct supervision? A principle motivation behind this project is to avoid direct supervision whenever possible and instead allow the system to learn the natural constraints which are implicit in the language. We hope to avoid direct supervision techniques in part by allowing the learning system to learn the majority of syntactic and contextual constraints from examples of correct sentences. This approach has been applied with some success in the past (e.g. Hanson and Keg's PARSNIP system) and is a promising avenue for avoiding excessive supervision.

The architecture of the learning system is a second major consideration. This involves both the nature of the architecture (is it static or dynamic) as well as the learning rule(s) associated with it. We have considered various basic recurrent network architectures such as Elman's simple recurrent network (SRN), Jordan's recurrent network and Miikkulainen's FGREP network. These architectures have been extensively utilized in the cognitive science literature, and we are considering various extensions to them. We hope to generalize these architectures to include multiple "memory" layers, as well as understand how these architectures can be combined together, perhaps even dynamically.

A final major consideration regards structured concept acquisition. Adding recurrence to the network is the principle method of allowing the networks distributed representation to exhibit structured concepts. Thus, this issue may reduce to issues regarding network architecture. Even so, concept acquisition remains a difficult problem, especially in the context of language acquisition. We feel that neural networks can develop and utilize structured concepts in a manner which is quite different from traditional concept acquisition. For example, we expect that it will be easier to combine different aspects of textual comprehension (e.g. parsing, pronominal reference, ambiguity resolution) with a distributed representation than it has been with both symbolic and localist-connectionist systems.

At the moment, this research has yet to generated any publishable material. Unfortunately, my work at Peregrine Systems is my first exposure to research in natural language processing so I have no other related publications. However, I do have a substantial background in topics related to natural language processing. I have significant academic experience in Cognitive Science. This includes courses in human learning and inductive processes, as well as language acquisition in children. The later provided an excellent contrast between Chomsky's theoretical model of language acquisition to Piaget's developmental model.

Additionally, my current academic research interests involve machine learning and adaptive computation. I am well versed in the issues involved with learning tasks in a variety of domains, including neural networks, traditional machine learning (e.g. Samuel's checker player) and others. I have also studied Valiant's learning theory model extensively and am currently applying this analysis to genetic algorithms (learning algorithms based on principles of evolution and selection).

In summary, I believe I would prove to be a productive member of this workshop. My research relates to many of the issues considered by the workshop, and I am prepared to make interesting contributions to the discussing of the topics covered. My background in cognitive science machine learning and natural language development is sufficiently well developed to enable me to intelligently consider the issues at hand.

# Automatic Acquisition of Word Meanings

Peter M. Hastings
EDS Center for Machine Intelligence
Ann Arbor MI 48105
and
Artificial Intelligence Laboratory
The University of Michigan
Ann Arbor, MI 48109

Steven L. Lytinen
Artificial Intelligence Laboratory
The University of Michigan

## 1 Introduction

We present an incremental approach to the task of learning words from context. The learning task is defined as follows: given a set of natural language sentences in which a previously unknown lexical item appears, infer the syntactic class and the meaning (or meanings) of the word. We assume that the vast majority of other words appearing in the set of sentences are already known.

Our approach has been implemented as part of a natural language processing system called LINK (Lytinen, 1990; Lytinen, in press). LINK uses a unification grammar and integrated syntactic and semantic processing. We are using LINK in two prototype applications involving relatively narrow domains (i.e. the necessary domain knowledge can be described fairly completely), but the textual input is entered by a large number of users and is therefore subject to wide variations in the terminology used. Our system is able to infer the meanings of many unknown words in these applications.

Although our approach is used to infer both the syntactic category and the meaning of unknown words, we will only discuss the learning of meanings in this paper. The reader is referred to (Lytinen and Roberts, 1988) for a discussion of syntactic learning in LINK.

## 2 The Approach

LINK's domain knowledge is organized in a simple IS-A hierarchy. For each concept in the hierarchy, we define a set of thematic roles or "slots" that can be attached to the concept, as well as the type of concept which can fill each slot. The set of restrictions on fillers of slots for a concept must be at least as specific as the restrictions for its ancestors in the hierarchy (i.e. more general concepts).

Figure 1 presents an example hierarchy, taken from one of our two prototype domains. Texts in this domain describe sequences of activities to be performed on an assembly line. In this hierarchy, since ACTION requires an ACTOR that is ANIMATE, this restriction also implicitly holds for REPAIR-ACTION, ADJUST-ACTION, and all other descendants of ACTION. CALIBRATE is an example of a concept which makes a further restriction on a previously constrained slot. Since ADJUST-ACTION requires an OBJECT which is a DEVICE, the additional restriction on this slot under CALIBRATE must be a descendant of DEVICE.

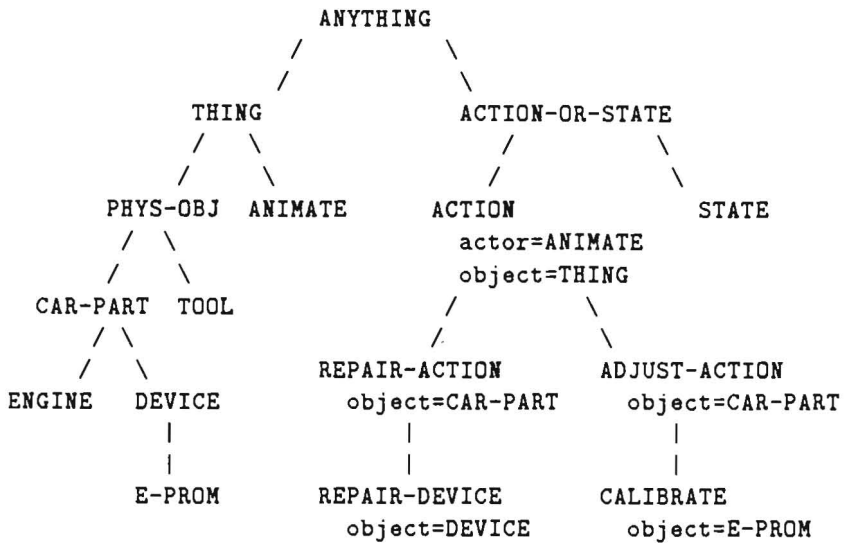



Figure 1: A simple concept hierarchy for LINK

LINK's domain knowledge is used in the process of learning word meanings. Initially, it is assumed that every concept in the hierarchy is a candidate hypothesis for the meaning of an unknown word. Example sentences can provide two types of restrictions on the set of candidate hypotheses. First, the unknown word may appear as the filler of a thematic role of another word. For example, in the sentence "Calibrate the flarge," LINK's unification grammar suggests that "flarge" is the semantic OBJECT of CALIBRATE. This condition places an upper bound on the generality of the word's meaning: "flarge" must be an E-PROM or a descendant of E-PROM in the hierarchy, since only E-PROM's can be CALIBRATEd. Second, context may suggest a filler for a thematic role of the unknown word, as in the sentence "Flarge the engine." In this case, LINK's unification grammar suggests that ENGINE is the semantic OBJECT of "flarge." Information about role-fillers of an unknown concept place a lower bound on the specificity of the concept: given that ENGINE is the OBJECT, "flarge" cannot refer to a concept that is lower in the hierarchy than REPAIR-ACTION or ADJUST-ACTION, since concepts below this in the hierarchy do not allow ENGINEs to be their OBJECTs.

Given that these two types of restrictions are provided by example sentences, this would suggest a least-commitment approach to learning, such as Mitchell's *candidate-elimination* algorithm (Mitchell, 1977). Mitchell's algorithm used *version spaces* to represent the set of candidate hypotheses, and slowly narrowed the version space depending on the additional constraints provided by new examples. Unfortunately, in our word learning task, often it is the case that particular kinds of words only appear in examples that provide one of the two types of restrictions. Nouns, which usually refer to things, almost always appear as role-fillers of actions or states; thus, examples only serve to limit the upper bound of the candidate hypotheses. Verbs, on the other hand, usually appear with role-fillers attached to them, and not as role-fillers themselves, since they refer to actions or states. Thus, examples only serve to place a lower bound on their candidate hypotheses. Thus, since examples only provide one of the two kinds of restrictions for many word classes, a least-commitment algorithm would not converge on a single hypothesis for the meaning of many unknown words.

Because of this, our algorithm is not a least-commitment algorithm. For nouns, we assume the most

general candidate hypothesis is the correct one. Thus, the hypothesis for "Calibrate the flarge" is that "flarge" means E-PROM. In the case of verbs, the most specific candidate hypotheses are kept. From "flarge the engine," then, "flarge" is assumed to mean either REPAIR-ACTION or ADJUST-ACTION. A later example like "flarge the wrench" would cause generalization to occur, since a box cannot be the object of either REPAIR-ACTION or ADJUST-ACTION. A search is initiated up the hierarchy from these concepts until a concept or set of concepts is found that can take both ENGINEs and TOOLs as objects. In this case, the new hypothesis would be that "flarge" means ACTION.

# 3 Limitations of This Technique

The learning mechanism described here is not suggested to be a solution to the problem of automatic acquisition of word meaning. Rather, it is an attempt to demonstrate how the use of a small amount of semantic information that is required in the performance of the parsing process along with a general-purpose learning algorithm can make major strides toward inferring a useful word meaning hypothesis. Several artifacts of the learning mechanism limit what can be learned.

The first is the assumption that the representation of the ontology is complete, that is that every concept which is part of the domain is *a priori* represented by some node in the semantic hierarchy. This clearly limits the range of concepts that can be learned.

In addition, this techniques relies solely on one type of information, the semantic constraints of role-fillers. While this information is sufficient to differentiate between many of the word meanings, large classes of words exist that require additional information to distinguish the members of the class.

As mentioned above, the learning algorithm can not handle ambiguous words. In such cases, an apparent contradiction is found between competing hypotheses, and an over-general concept is then chosen. Some sort of mechanism is needed to determine when a more general concept is required or when a disjunctive mapping is justified.

Finally, the learning algorithm as we have described it so far often does not converge on a single hypothesis for the meaning of a word, especially in the case of verbs. To see this, consider the hierarchy in figure 2. It is the same as in figure 1, but with the additional action PICK-UP added. With this hierarchy, if the system is presented with an example such as "Flarge the e-prom," intuitively it seems that the best hypothesis for the meaning of "flarge" would be CALIBRATE, since only E-PROMs can be calibrated. However, other hypotheses cannot be eliminated as possibilities: "flarge" might mean REPAIR-DEVICE, since e-proms are also devices; and it might mean PICK-UP, since e-proms are also physical objects. Given the hierarchy as it stands, no examples can be given which will narrow down this set of candidate hypotheses (assuming "flarge" really does mean CALIBRATE), since nothing which meets the restrictions on the slots of CALIBRATE will violate any of the restrictions on the slots of REPAIR-DEVICE or PICK-UP. Concepts like PICK-UP, which have rather general restrictions on their slots, will be candidate hypotheses for the meanings of a relatively large number of unknown words, since often it will be the case that no examples are possible which will eliminate it from the list of candidates.

To remedy this problem, our algorithm ranks the list of candidate hypotheses according to how "tightly" each candidate's constraints on slots match with the actual slot fillers found in the examples. For the example "flarge the e-prom," CALIBRATE is the highest-ranked candidate hypothesis for the meaning of "flarge," since its restriction on the OBJECT slot exactly matches the OBJECT of "flarge" in the example sentence. REPAIR-DEVICE is ranked second and PICK-UP third, since DEVICE is closer in the hierarchy to E-PROM than PHYS-OBJ is.

# 4 Related Work

Gleitman (1990) proposed a mechanism called "syntactic bootstrapping" that children might use to guide their search for meanings of verbs through the space of possible meanings that could be inferred from the

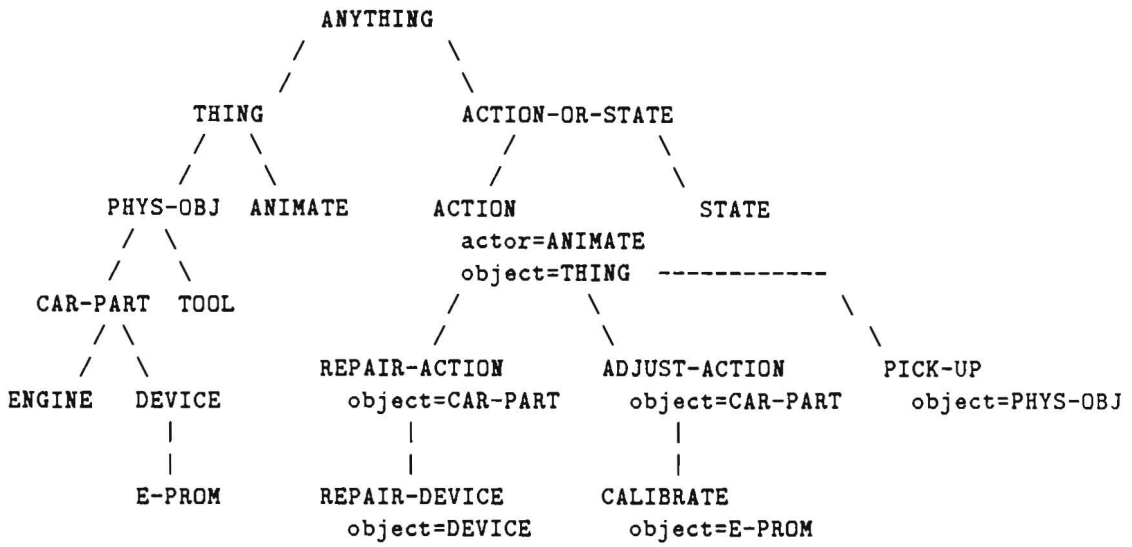



Figure 2: A slight variation of the first concept hierarchy

immediate context. She suggested that children as young as 17 months have the strong capabilities for recognizing syntactic distinctions and using them to constrain the meanings of verbs they are learning. For example, children who didn't know the meaning of the word FLEX were shown two videos, one of Big Bird and the Cookie Monster crossing and uncrossing their own arms, and another with one of them crossing the arms of the other. When the sentences *Big Bird is flexing with the Cookie Monster* and *Big Bird is flexing Cookie Monster* were broadcast through a speaker, the children showed a definite preference for the "syntactically congruent screen", i.e. the video that was showing the action that was being described, even though they had no semantic knowledge of the meaning of FLEX. Gleitman argued that without such a constraining mechanism, the task of word learning would be computationally infeasible. But while her approach relies solely on the syntactic structure of the sentence to yield semantic clues, our approach combines use of syntactic and semantic information (but no external context) to generate hypotheses.

Similar efforts at using machine learning techniques in lexical acquisition were reported in (Zernik, 1987). Zernik described his approach as using a version space technique to learn phrasal lexicon rules. However, Zernik's system receives feedback from a teacher in the form of user-supplied "contexts" that explain what the input means. It is not clear if Zernik's approach can be adapted to a situation in which feedback is not available.

Selfridge's CHILD program (1986) used contextual information to provide constraints on definitions of undefined words in much the same way as our system does for nouns. However, CHILD learned from only one example, and could not further refine meanings based on subsequent examples.

Jacobs and Zernik (1988) describe the RINA system, in which a task very similar to our word learning task is performed. RINA examines large corpora, extracting many examples of a given unknown word. Although they do not describe their algorithm in detail, it appears from examples discussed in the paper that word meaning acquisition in RINA is driven more heavily by discourse context than in LINK.

# 5 Future Work

There are many ways in which our algorithm can be extended. First, the algorithm as it currently stands only uses information about semantic dependencies that the parser is able to identify between words in example sentences. It should be able take advantage of other information available from the examples, such as the syntactic constructions used with an unknown word, additional semantic contextual information, and so on. We plan to investigate incorporating the use of some of this additional information into our learning algorithm.

Second, the assumption that a word must refer to a unique concept in the hierarchy is not a realistic one. Many words are ambiguous, and thus refer to two or more nodes in the hierarchy. Even an unambiguous word's meaning may not correspond exactly to an already existing node in the hierarchy. Our system should be able to entertain disjunctive hypotheses for word meanings, and should also be able to consider "splitting" a node in the hierarchy, so that a word can refer to a new subconcept.

Finally, we plan to test our algorithm in our two prototype domains to see how well it learns. We are currently testing our hypothesis ranking system to see how well it chooses the correct hypothesis for an unknown word from the list of candidate hypotheses. As we modify our algorithm further, testing will provide valuable feedback for us to see if our system's performance is improving.

## References


Gleitman, L. (1990). The structural sources of verb meanings. *Language Acquisition, I*(1), pp. 3-55.

Jacobs, P. and Zernik, U. (1988). Acquiring lexical knowledge from text: A case study. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, Minneapolis, MN, pp. 739-744.

Lytinen, S. (1990). Robust processing of terse text. In *Proceedings of the 1990 AAAI Symposium on Intelligent Text-based Systems*, Stanford CA, March 1990, pp. 10-14.

Lytinen, S. (in press). A unification-based, integrated natural language processing system. To appear in *Computers and Mathematics with Applications.*

Lytinen, S., and Roberts, S. (1989). Lexical acquisition as a by-product of natural language processing. In *Proceedings of the First International Lexical Acquisition Workshop*, IJCAI-89, Detroit, MI, August 1989.

Mitchell, T. (1977). Version spaces: A candidate elimination approach to rule learning. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pp. 305-309.

Selfridge, M. (1986). A computer model of child language learning. *Artificial Intelligence, 29*, pp. 171-216.

Zernik, U. (1987). Language acquisition: Learning a hierarchy of phrases. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pp. 125-132.

# Hybrid Models of Natural Language Learning

**Jane C. Hill**
**Department of Mathematics and Computer Science**
**Dickinson College**
**Carlisle PA 17013**

My research is involved with the building of computational models of acquisition of syntax. The first model [Hill '82] was a "schema theoretic" model that learned by adjusting weights in a semantic net representations of grammar, lexicon and world knowledge. Our approach has remained a minimalist approach, as opposed to innatist, since it is our feeling that it is more interesting to explore the limits of language that can be learned given general cognitive strategies and some world knowledge, than to begin by assuming that a great deal of specific linguistic knowledge must be built into the model. The model is intended for use in cognitive exploration. In our model the learning is implemented by adjusting weights in the semantic network representations. World knowledge must be given the model in order for even primitive syntactic knowledge to be induced. The system acquires a rudimentary semantic parser that is highly dependent upon the input data given. It learns both to understand input sentences and to generate responses. Current research in connectionist models of language acquisition and discussions of hybrid models have led us to reexamine the model with the idea of implementing various aspects of the model in a connectionist framework. We are interested in developing methodologies for the combining of connectionist modelling with more symbolic modelling to take advantage of the differing strengths of each approach.

**Bibliography**

J.C.Hill, Language Acquisition in Encyclopedia of Artificial Intelligence, second edition, John Wiley & Sons, to appear. (original article, first edition, 1987.)

M.A. Arbib & J.C.Hill, Language Acquisition: Schemas Replace Universal Grammar, Chapter 3 in John H. Hawkins, ed., Explaining Language Universals, Basil Blackwell, 1988, pp.56-101.

M.A.Arbib, E.J.Conklin, & J.C.Hill, From Schema Theory to Language, Oxford University Press, 1987.

J.C.Hill, Using a Computational Model of Language Acquisition to Address Questions in Linguistic Inquiry, Proceedings of the Seventh Annual Conference of the Cognitive Science Society, University of Massachusetts at Amherst, August 1986, 407-419.

J.C.Hill, Using a Computational Model of Language Acquisition to Address Questions in Linguistic Inquiry, Proceedings of the Seventh Annual Conference of the Cognitive Science Society, University of California at Irvine, August, 1985, 298-302.

J.C.Hill & M.A.Arbib, Schemas, Computation, and Language Acquisition, Human Development, 1984, 27 (5-6), 282-296.

J.C.Hill, Combining Two-Term Relations: Evidence in Support of Flat Structure, Journal of Child Language, 1984, 11(3), 673-678.

J.C.Hill, A Computational Model of Language Acquisition in the Two-Year-Old, Cognition and Brain Theory, 1983, 6(3), 287-317.

J.C.Hill, A Computational Model of Language Acquisition in the Two-Year-Old, University of Massachusetts at Amherst, Ph.D. Dissertation, September 1982, reproduced by the Indiana University Linguistics Club, Bloomington, Indiana, February 1983.

## Hybrid Models of Natural Language Learning

One way to contrast symbolic models of language learning with connectionist models is to observe differences in "grain size". While symbolic models typically examine a large picture e.g, Minsky's frames or Schank's scripts, connectionist models typically examine in fine detail some aspect of a larger problem- -as for example McClelland Rumelhart's parallel distributed processing model of the learning of the past tense of verbs in English. Typically our "schema-theoretic" models lie at some midpoint between the two. A schema is a unit belonging to an internal model of the world that may vary in grain size since we may have a schema for an object or for a detail of an object. Schemas are dynamic and may combine to form new schemas. We originally described our schema-theoretic models as being "in the style of the brain". While a schema-theoretic model may be more symbolic than connectionist, it is constrained to be at a sufficiently low level that in principle its separate parts could be instantiated in terms of a neural network. Schema-level formalisms, however, only approximate the behaviour of a model expressed in neural net formalism. Unlike more traditional symbolic models, however, a schema-theoretic model is composed of many redundant and overlapping modules -- one of the reasons that such models are described as being in the style of the brain. These observations may be made more explicit by contrasting two models

that learn the past tense of verbs in English. The first is the well-known McClelland and Rumelhart connectionist model "On Learning the Past Tenses of English Verbs" [Parallel Distributed Processing, 1986, vol.2, 216-268], and the second is my schema theoretic model [Hill 1986]. Whereas McClelland and Rumelhart modelled only the learning of the past tense forms of verbs, we were able to model their learning within a larger framework. This is as one would expect because of the typical difference in grain-size mentioned above. The input to the McClelland and Rumelhart model was 420 verbs. Input to our model was simply adult sentences. This is important, since McClelland and Rumelhart have been criticized for presenting a small set of common verbs to their model before presenting the remaining verbs. Our model achieves the same kind of selectivity, but with rules of salience. Our model, given a body of sentences, attends first to a set of simple verbs. McClelland and Rumelhart's input was phonologically encoded by means of sets of Wickelfeature units. Our model simply glossed over the phonological nature of the input. It was because of the precise nature of the input that their model could make the detailed predictions that they made. Yet it was also the Wickelfeature representation that suffered the brunt of the criticism, notably from Pinker and Prince ["On language and Connectionism: Analysis of a Parallel Distributed Processing Model of Language Acquisition" in Connections and Symbols, A Special Issue of Cognition, Bradford/MIT 1988]. Detailed predictions will foment detailed objections. Yet in their model the most important claim was not questioned, namely that systems lacking rules can exhibit rule-like behavior.

The primary advantage of their model is in terms of the detail of the observed behaviour. Both models capture the basic three-step path of acquisition of past tense forms, but their model can show a correlation between difficulty of learning particular past tense forms in the model and in the child. They capture many aspects of the differences in performance on different types of verbs. Our model can draw no such detailed conclusions about the difficulty of learning a specific verb. What is learned or not learned in our model depends entirely upon the input corpus. Typically the connectionist model is rich in specific predictions.

The connectionist model is computationally intensive. They used a single long run of 260 hours to get their results. Our model is a vehicle for experimentation. One can select a body of sentences and give them to the model and watch them run, or alternatively one can sit and type a set of sentences into the model and see what comes out. Some type of response is instantaneous. One can draw conclusions about what the model attends to depending upon the current stage of the model's grammar. We agree in our aim to address the time course of mental development. Their model required a specific teaching phase followed by a testing phase. Our model is more realistic in that teaching and testing happen in parallel. We neither of us require explicit representation of a general rule, but instead make use of a decentralized interaction of many components to yield behavior that is describable by a rule, but in no way is the expression of a rule. Our feeling is that at this early stage of building models it is important to continue to pursue the implementation of all kinds of models because each variety has its strengths and its weaknesses. Our current work is too incomplete to merit anything more that a brief comment. We are currently building a model to acquire the rules of -un prefixation. Linguists [M. Bowerman, for one, in "The No Negative Evidence Problem" in Explaining Language Universals, Hawkins, ed., Basil Blackwell, 1983] have searched for and failed to find a set of rules that describe those verbs which can be prefixed by -un. Yet adults have no difficulty in recognizing inappropriate use of the prefix. The lack of a simple coherent set of rules has led us to implement a connectionist model to explore the discrimination between verbs which may be prefixed by -un and those that may not. Our first version of the model uses a simple back-propagation learning paradigm over a set of features. It is our intention to match the output of the model against instances of the use of un- by children in the CHILDES data base that is maintained at Carnegie-Mellon University. Ultimately we hope to incorporate this learning paradigm within the larger framework of our schema- theoretic model.

# Toward Integrated Models of Natural Language Evolution, Development, Acquisition, and Communication In Multi-Agent Environments

Vasant Honavar
Department of Computer Science
226 Atanasoff Hall
Iowa State University
Ames, Iowa 50011, U.S.A.

First draft November 21, 1990; Revised January 31, 1991

## 1. Motivation

The evolution of natural language in communities of individuals over several generations, the development of language over time, the acquisition of language by language-naive individuals born into the community or introduced into it from outside, the role played by language in communication among individuals, and the influence of the communicative role of language on the evolution, development and acquisition of language are among some of the central questions in cognitive science and artificial intelligence. The dominant tendency in these fields over the past several decades has been to approach the study of perception, language, problem solving, motor behavior as if they were isolated from one another. For instance, computer scientists who work on computational models of vision (with applications in artificial intelligence and robotics, neuroscience and cognitive science) and psychologists who work on visual perception have little or nothing to do with computer scientists who work on natural language and psychologists and linguists who study language behavior. The study of learning is, for the most part, just as isolated. Learning research often proceeds as if the content of what is learned has little bearing on the basic processes postulated. This is especially true of most work in computational approaches to the study of learning.

In contrast, the distributed artificial intelligence framework outlined below attacks the problem of understanding natural language evolution, development, acquisition, and communication processes as an integral part of the broader task of understanding the design, function, development, adaptation, and evolution of cognitive agents.

## 2. General Framework

A primary function served by language (defined broadly) is communication among individuals. It is therefore reasonable to address language acquisition in the context of a group of individuals which have to communicate with each other in order to survive. Within

This work has been influenced by several authors - J. Hattiangadi, C. Peirce, J. Piaget, L. Vygotsky, R. Allen, R. Narasimhan, L. Uhr, P. Langley, J. Laird, S. Harnad, M. Minsky, M. Dyer, and G. Lakoff - among others. Their influence is acknowledged without in any way implicating them in the views expressed in this abstract.

this paradigm, the group shares a common *environment*. Each individual or *agent* is endowed with one or more sensory channels (e.g., visual, auditory) through which it receives environmental *stimuli*. Each agent has its disposal, a set of primitive *actions* that enable it to manipulate (e.g., grasp, move) objects in its environment. It can transmit *signals* or *symbols* that impinge on the sensory channels of other individuals. Each agent also has associated with it, a set of *goals*, *internal drives*, and *needs* (e.g., hunger). The environment embodies complex dynamics: objects in the environment can change their states in an orderly or disorderly manner. Each such state change constitutes an *event*. Each agent might sense events (through its sensory channels), respond to events (using its repertoire of actions), or communicate about events to other individuals (by transmitting appropriate signals or symbols).

Several interesting questions can be raised within this framework:

(1) Given a subset of individuals that use a predetermined language, what basic mechanisms are needed for individuals newly introduced into the environment to learn to communicate using the language of the community? This situation is akin to language acquisition by children. Children's competence in naive observation and the use of ordinary language improves as they grow and enables them to acquire more or less dependable knowledge of the environment in which they live. A model of language behavior must come to grips with the developmental issues involved in the process:

Language acquisition in children appears to be incremental. This constrains our model of language acquisition to relying on incremental learning methods. It also appears to proceed in discernible developmental stages. *Generative learning* structures and processes (Honavar & Uhr, 1991) are being applied in this context.

Language learning entails acquisition of *meanings* which involves a mapping of words or sentences into the corresponding sensory, motor, or internal representations and vice versa; The acquisition of the abilty to judge the grammaticality of sentences in and of itself does not amount to language acquisition. Furthermore, such meanings are acquired in a *social* context, i.e., within a community of multiple agents which have to communicate among themselves to attain their goals. *Generative learning* structures and processes (Honavar & Uhr, 1988; Honavar, 1990; Honavar & Uhr, 1991) enable agents to *extract*, *abstract*, and *encode* multi-modal sensory and action patterns and associations among them. Internal representations of the environment so developed derive their meanings by virtue of being *grounded* in corresponding analogical sensory (e.g., visual, auditory) and motor representations. Meanings of such internal representations are further enriched by the role they play in communication among individuals in a community.

(2) A variant of the scenario outlined above is one in which individuals that are already proficient in communicating in some language (first language) are introduced into a community that uses a different language (second language). We can then ask, what basic processes are used by such individuals in learning the second language? We can also study the effect that the knowledge of the first language has on the learning of the second language; the effect of learning the second language on communication in the

first language, and so on. The results of such computational models could be correlated with empirical data on second language learning to identify the structures and processes that account for such data and to suggest further empirical studies.

(3) Assume two isolated communities, each with a distinct language that inhabit more or less identical environments. We can examine a scenario in which two such communities are brought into contact with each other. This places a demand on individuals across both communities to communicate with each other. The changes observed in the individual languages or the birth of a new language through such interaction can offer insights into evolution of new languages from old ones.

(4) Assume a community of individuals into which new language-naive individuals are introduced over time at intervals determined by a birth-rate and in which a fraction of the individuals perish at intervals determined by a death-rate. Assume that the environment and/or the sensory channels, primitive actions, internal needs and drives change over time at a rate slower than the birth or death rates. We can then ask what sort of changes manifest themselves in the language used by the community over time? This is akin to the gradual changes in natural language used by communities over time and is useful in modeling such changes.

(5) Consider the scenario in which we start with a community of language-naive individuals. We can then ask what basic mechanisms are necessary to ensure that such a community over time evolves a sufficiently powerful language that can then be readily learned by language-naive individuals as they are introduced into the community. This scenario corresponds to the biological evolution of language. Adaptations of evolutionary learning methods are being investigated in this context.

## 3. Modelling Agents, Objects, and Environments

We are in the process of developing software tools that would facilitate the computational modelling of scenarios of the sort outlined above. Object-oriented programming paradigm (e.g., CLOS) with its loosely organized collection of interacting entities provides a versatile tool for building such tools.

We have chosen to model each individual in our toy *language community* by a *generalized connectionist network* (GCN) (Honavar & Uhr, 1990f). GCN offer many extensions - especially in the form of *generative* or *constructive* learning structures (Honavar & Uhr, 1991), and coordination and control structures (Honavar & Uhr, 1990a) to the currently popular *connectionist network* (CN) models. GCN offer an attractive and versatile framework for the integration of connectionist network and symbol processing approaches to the modelling of intelligent systems. They also facilitate an exploration of the tradeoffs between parallel versus serial computation, local versus distributed processing, memory, and control, symbolic versus sub-symbolic representations, etc.

Very briefly, a GCN is a graph (of linked nodes) with a particular topology $\Gamma$. The total graph can be partitioned into three functional sub-graphs - $\Gamma_{\mathbf{B}}$ (the *behave/act* sub-graph), $\Gamma_{\Lambda}$ (the *evolve/learn* sub-graph), and $\Gamma_{\mathbf{K}}$ (the *coordinate/control* sub-graph). The nodes in a GCN compute one or more different type/s of functions: **B** (*behave/act,*); $\Lambda$ (*evolve/learn,*);

and **K** (*coordinate/control*).

$$GCN = \{\Gamma, \mathbf{B}, \Lambda, \mathbf{K}\}$$

Starting with a very simple caricature, the model of an agent is being elaborated as necessary by adding simple planning capabilities, increasingly sophisticated inductive and deductive learning capabilities, increasingly intricate memory structures, and so on. The strategy being adopted is one of incremental development aided by exploratory programming wherein new capabilities are added only when indicated by extensive simulation studies.

The environment in which the agents interact is modelled as a collection of objects that obey certain *physical laws*. Agents typically have no built-in explicit knowledge of such laws; They can however, develop (necessarily incomplete) internal models of their environment through learning (by direct interaction with the environment or through communication with other agents in the community).

## 4. Some Areas of Current Emphasis

The general framework sketched out above is obviously very broad. Our efforts at present are directed toward a few specific issues:

[1] Development of structures that model *internal drives*, *needs*, and *goals* of agents in a flexible manner that would facilitate us to study the interaction between such structures and adaptation and learning processes (e.g., generative learning (Honavar & Uhr, 1991)).

[2] Development of a framework that integrates processes on an evolutionary time-scale (e.g., competition, selection, cooperation) that operate over several generations of agents in our toy environment. Here we are influenced considerably by on-going work by several researchers in *artificial life* and evolutionary learning algorithms.

[3] Development of a simple psychologically motivated model of language acquisition by infants. The focus here is on identifying a small set of computational structures and environmental influences (e.g., the nature and extent of interactions with adults - agents already proficient in the language - e.g., verbal and non-verbal feedback) that are necessary and sufficient to facilitate such language acquisition. Our long-term goal is to use the results of this modelling effort along with data from developmental studies of children published in the psychology literature to guide further refinement of the model to examine language acquisition in increasingly complex environments.

[4] Development of models that can successfully demonstrate the acquisition of relatively simple languages with a pre-defined structure (e.g., languages that describe a small set of spatial and temporal relationships among simple geometrical objects) by language-naive individuals placed in a community of agents that use such a language to communicate with other agents.

## 5. Summarizing Comments

Languages evolve to meet the needs of individuals and communities; The primary role of language is communication among individuals dictated by their internal needs and

environmental pressures for survival; We believe it is impossible to approach the study of language in isolation from processes such as learning, perception, and cognition that individuals engage in within the context of a larger environment. We have outlined a general framework for study of natural language communication, acquisition, development and evolution in communities of interacting agents. We have also sketched out some tentative steps we have taken toward the construction of such a framework using an object-oriented programming paradigm. Much work remains to be done.

## 6. Related Publications


Honavar, V., & Uhr, L. (1987). *Recognition Cones: A Neuronal Architecture for Perception and Learning*. Technical report 717. Madison, WI: Computer Sciences Dept.

Honavar, V., & Uhr, L. (1988). A network of neuron-like units that learns to perceive by generation as well as reweighting of its links. In G. E. Hinton, T. J. Sejnowski, & D. S. Touretzky (Eds.) *Proceedings of the 1988 Connectionist Models Summer School*. San Mateo, CA: Morgan Kaufmann.

Honavar, V., & Uhr, L. (1989a). Brain-Structured connectionist networks that perceive and learn. *Connection Science - Journal of Neural Computing, Artificial Intelligence and Cognitive Research* 1 139-160.

Honavar, V., & Uhr, L. (1989b). Generation, local receptive fields and global convergence improve perceptual learning in connectionist networks. In *Proceedings of the 1989 International Joint Conference on Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann.

Honavar, V. (1989). *Perceptual development and learning: From behavioral, neurophysiological, and morphological evidence to computational models*. Technical report 818. Madison, WI: University of Wisconsin, Computer Sciences Dept.

Honavar, V. (1990). *Generative Learning Structures for Generalized Connectionist Networks*. Unpublished Ph.D Dissertation, University of Wisconsin-Madison.

Honavar, V., & Uhr, L. (1990a). Coordination and control structures and processes: possibilities for connectionist networks. *Journal of Experimental and Theoretical Artificial Intelligence* (In Press).

Honavar, V., & Uhr, L. (1990b). Efficient learning using multi-resolution representations of spatial, temporal, and spatio-temporal patterns. In *Proceedings of the 1990 Indiana-Purdue Conference on Neural Networks* (In Press).

Honavar, V., & Uhr, L. (1990c). Analog and hybrid analog-digital microcircuits for connectionist networks. (Paper in preparation).

Honavar, V., & Uhr, L. (1990d). On the role of reciprocal links in neural networks. (Paper in preparation).

Honavar, V., & Uhr, L. (1990e). *Symbol Processing Systems, Connectionist Networks, and Generalized Connectionist Networks*. Technical report #90-24. Ames, IA: Iowa State University, Department of Computer Science.

Honavar, V., & Uhr, L. (1991). Generative Learning Structures and Processes for Generalized Connectionist Networks. Technical report #91-02. Ames, IA: Iowa State University, Department of Computer Science.

# Restrictions on Grammar Size in Language Identification (Extended Abstract)

Sanjay Jain
Dept. of CIS
University of Delaware
Newark, DE 19716

Arun Sharma *
Dept. of Brain and Cog. Sci.
MIT
Cambridge, MA 02139

**Research Interests and Representative Bibliography**

Our research interests include Machine Learning and Structural Complexity.

[JS91a] *Learning with the Knowledge of an Upper Bound on Program Size*, by Sanjay Jain and Arun Sharma. To appear in Information and Computation.

[JS91b] *Learning in the Presence of Partial Explanations*, by Sanjay Jain and Arun Sharma. To appear in Information and Computation.

[JS90a] *Language Learning by a "Team"*, by Sanjay Jain and Arun Sharma. In the proceedings of International Colloquium on Automata, Languages and Programming, 1990.

[JS90b] *Finite Learning by a "Team"*, by Sanjay Jain and Arun Sharma. In the proceedings of Computational Learning Theory, 1990.

[JS90c] *Characterizing Language Learning by Standardizing Operations*, by Sanjay Jain and Arun Sharma. In the proceedings of International Conference on Computing and Information, 1990.

[FJ90] *Approximate Inference and Scientific Method*, by Mark Fulk and Sanjay Jain. In the proceedings of Algorithmic Learning Theory, 1990.

[CJS90] *Anomalous Learning Helps Succinctness*, by John Case, Sanjay Jain and Arun Sharma. In the proceedings of Algorithmic Learning Theory, 1990.

[FJ89] *Learning in Presence of Inaccurate Information*, by Mark Fulk and Sanjay Jain. In the proceedings of Computational Learning Theory, 1989.

[CJS89] *Convergence to Nearly Minimal Size Grammars by Vacillating Learning Machines*, by John Case, Sanjay Jain and Arun Sharma. In the proceedings of Computational Learning Theory, 1989.

**Abstract**

Study of program size restrictions in inductive learning is motivated with arguments from "formal language learning theory" and "computational philosophy of science". A number of identification criteria resulting from various size restrictions on programs inferred in the limit by an inductive inference machine are considered. A main concern of the paper is the investigation of relationships of these criteria with *acceptable* programming systems.

*Supported by a grant from the Siemen's Corporation

# 1 Introduction

Motivated by psycholinguistic studies which conclude that children are rarely, if ever, informed of grammatical errors, Gold [Go67] introduced the seminal notion of identification as a model for first language acquisition. According to this paradigm, a child (modeled as a machine) receives (in arbitrary order) all the well-defined sentences of a language, and simultaneously conjectures a succession of grammars. A criterion of success is for the child to eventually conjecture a correct grammar for the language being received and never to change its conjecture thereafter. Replacing the child machine by an arbitrary machine in this scenario yields a formal model of language acquisition. This model is essentially Gold's influential language learning paradigm discussed, for example, by Pinker, Wexler and Culicover, Wexler, and Osherson, Stob, and Weinstein. However, Gold's paradigm is a highly idealized model which assumes unbounded resources in the form of time and storage. In the present paper, we investigate restrictions on the above criterion of successful learning where a machine is required to conjecture "succinct" grammars. The main results of this paper demonstrate that such restrictions result in learning criteria that are dependent on the choice of programming system used to interpret a machine's conjectures. Our treatment is recursion theoretic and some of our results build on results and techniques from inductive inference of recursive functions studied by Freivalds, Kinber, Chen and Case, Jain, and Sharma.

In section 2 we introduce the notation and the preliminary notions of language, grammar, and programming system. In section 3 we describe Gold's paradigm and observe that classes of languages that can be learned are independent of the choice of programming system used to interpret machines' conjecture. In section 4 we introduce a number of restrictions in Gold's paradigm, which restrictions require that a machine converge to a "succinct" grammar. For each of these restrictions, we show that the classes of languages that can be learned is dependent on the choice of programming system used to interpret a machine's conjectures. Finally, section 5 contains a brief discussion of our results.

# 2 Notations

$N$ denotes the set of natural numbers, $\{0,1,2,3,\ldots\}$, and $N^+$ denotes the set of positive integers, $\{1,2,3,\ldots\}$. Generally, lower case letters near the beginning, middle, and end of the alphabet, with or without decorations, $a,b,c,\ldots,i,j,k,l,m,n,\ldots,x,y,z$, range over $N$.

$\in$, $\subseteq$, and $\subset$ denote, respectively, membership, containment, and proper containment for sets (including sets of ordered pairs). We let $P,S$, with or without decorations, range over subsets of $N$ and we let $D$, with or without decorations, range over finite subsets of $N$. $\|P\|$ denotes the cardinality of $P$. $min(P)$ and $max(P)$ respectively denote the minimum and maximum element in $P$. We take $min(\emptyset)$ to be $\infty$ and $max(\emptyset)$ to be 0. Let $\lambda x,y.\langle x,y\rangle$ denote a fixed pairing function (a recursive, bijective mapping: $N \times N \rightarrow N$) [Ro67]. $\lambda x,y.\langle x,y\rangle$ and its inverses are useful to simulate the effect of having multiple argument functions. $\pi_1$ and $\pi_2$ are corresponding projection functions, i.e., $(\forall x,y)[\pi_1(\langle x,y\rangle) = x \wedge \pi_2(\langle x,y\rangle) = y]$.

$L$, with or without decorations, ranges over *recursively enumerable* (r.e.) subsets of $N$, which subsets are usually construed as codings of formal languages. $\mathcal{E}$ denotes the class of all recursively enumerable languages $\subseteq N$. We let $\mathcal{L}$, with or without decorations, range over subsets of $\mathcal{E}$. $L_1 \triangle L_2$ denotes $(L_1 - L_2) \cup (L_2 - L_1)$, the symmetric difference of $L_1$ and $L_2$. $\eta$ and $\xi$ range over partial functions. domain($\eta$) and range($\eta$) respectively denote the domain and range of partial function $\eta$.

$\mathcal{R}$ denotes the class of all *recursive* functions, i.e., total computable functions with arguments and values from $N$. $f, g, h$, and $p$, with or without decorations, range over $\mathcal{R}$. $\mathcal{S}$ ranges over subsets of $\mathcal{R}$. $\mathcal{R}^+$ denotes the set of recursive functions with range a subset of $N^+$.

$\psi$ with or without decorations ranges over *acceptable programming systems* [Ro58, Ro67, MY78] for the partial recursive functions: $N \rightarrow N$. We let $\varphi$ to be a fixed acceptable programming system. $\psi_i$ denotes the partial recursive function computed by $\psi$-program number $i$. $W_i^\psi$ denotes domain($\psi_i$). $W_i^\psi$ is, then, the r.e. set/language ($\subseteq N$) accepted (or equivalently, generated) by the $\psi$-program number $i$. We let $\Psi$ be an arbitrary Blum complexity measure [Bl67b] associated with acceptable programming system $\psi$; such measures exist for any acceptable programming system [Bl67b]. For a given total computable function $f$ and an r.e. language $L$, we define $\mathbf{minprogram}_\psi(f) = min(\{i \mid \psi_i = f\})$ and $\mathbf{mingrammar}_\psi(L) = min(\{i \mid W_i^\psi = L\})$.

The quantifiers '$\forall^\infty$' and '$\exists^\infty$' mean 'for all but finitely many' and 'there exist infinitely many,' respectively. The quantifier '$\exists!$' means 'there exists a unique.' Any unexplained notation is from [Ro67].

# 3 Gold's Paradigm

In this section we briefly introduce Gold's paradigm for language learning. A *sequence* $\sigma$ is a mapping from an initial segment of $N$ into $(N \cup \{\#\})$. The *content* of a sequence $\sigma$, denoted by content($\sigma$), is the set of natural numbers in the range of $\sigma$. The length of $\sigma$, denoted by $|\sigma|$ is the number of elements in the domain of $\sigma$. A *text* T for a language $L$ is a mapping from $N$ into $(N \cup \{\#\})$ such that $L$ is the set of natural numbers in the range of $T$. The *content* of a text $T$, denoted by content($T$) is the set of natural numbers in the range of $T$. $T[n]$ denotes the finite initial sequence of $T$ with length $n$. Suppose M is a learning machine and $T$ is a text. $\mathrm{M}(T)\downarrow$ (read $\mathrm{M}(T)$ converges) iff $(\exists i)(\forall^\infty n)[\mathrm{M}(T[n]) = i]$. In this case we say that $\mathrm{M}(T)\downarrow = i$ (read M converges on $T$ to $i$).

**Definition 1** [Go67] Let $\psi$ be an acceptable programming system. A machine M $\mathbf{TxtEx}_\psi$ identifies $L$ (written: $L \in \mathbf{TxtEx}_\psi(\mathrm{M})$) iff ($\forall$ texts $T$ for $L$)$(\exists i \mid W_i^\psi = L)[\mathrm{M}(T)\downarrow = i]$.

Below we define the inferring power of the above criterion which is a set theoretic summary of the capability of various machines to learn according to the criterion.

**Definition 2** [Go67] $\mathbf{TxtEx}_\psi = \{\mathcal{L} \mid (\exists \mathrm{M})[\mathcal{L} \subseteq \mathbf{TxtEx}_\psi(\mathrm{M})]\}$.

**Proposition 1** *For all acceptable programmings systems* $\psi, \psi'$, $\mathbf{TxtEx}_\psi = \mathbf{TxtEx}_{\psi'}$.

Because of the above proposition we often refer to $\mathbf{TxtEx}_\psi$ (for an acceptable programming system $\psi$) by **TxtEx**.

# 4 Minimal Size Restriction

The size of the final stabilized grammar can be very "large." This poses a difficulty for Gold's paradigm to be a model of language acquisition. We describe this problem in the context of a child modeled as a machine. The human head is of bounded size. A simple result from computability

theory tells us that any recursively enumerable language can be generated by infinitely many syntactically distinct grammars whose size is bigger than any prespecified bound on the size of a child's head. A child learning a language, hence, must converge to a grammar which fits in its finite size head. This of course assumes that human brain storage is not magic, admitting of infinite regress, etc. An interesting complexity restriction to make, then, on the final grammar converged to in the limit is that it be of "small" size.

Notions from Blum [Bl67a] allow us to treat index for grammars as a program size measure. Our results can be suitably modified to hold for any other Blum size measure. A natural restriction, then, to make on the size of the final grammar is to require that it be of strictly minimal size. Definition 3 below describes this criterion.

**Definition 3**
(a) Let $\psi$ be an acceptable programming system. M $\mathbf{TxtMin}_{\psi}$-identifies $L$ (written: $L \in \mathbf{TxtMin}_{\psi}(\mathrm{M})$) iff $(\forall$ texts $T$ for $L)(\forall^{\infty} n)[\mathrm{M}(T[n]) = \mathbf{mingrammar}_{\psi}(L)]$.
(b) $\mathbf{TxtMin}_{\psi} = \{\mathcal{L} \mid (\exists \mathrm{M})[\mathcal{L} \subseteq \mathbf{TxtMin}_{\psi}(\mathrm{M})]\}$.

Surprisingly, as a contrast to Proposition 1 we have,

**Theorem 1** *There exist acceptable programming systems $\psi$ and $\psi'$ such that* $\mathbf{TxtMin}_{\psi} \neq \mathbf{TxtMin}_{\psi'}$.

Thus the classes of languages which can learned via minimal grammars depends on the programming system used to interpret the conjectures of the inference machine. Freivalds considered identification via programs which are of minimal size modulo a recursive (fudge) factor, i.e., the programs inferred are nearly minimal size. Case and Chi considered an analog of nearly minimal identification in the context of language learning. Definitions below describe this notion for language learning.

**Definition 4** [CC86] Let $h \in \mathcal{R}$.
(a) M $\mathbf{TxtMex}(h, \psi)$-*identifies* $L \in \mathcal{E}$ (written: $L \in \mathbf{TxtMex}(\mathrm{M}, h, \psi)) \Longleftrightarrow$ M $\mathbf{TxtEx}$-identifies $L$ in the acceptable programming system $\psi$ and $(\forall$ texts $T$ for $L)[\mathrm{M}(T) \leq h(\mathbf{mingrammar}_{\psi}(L))]$.
(b) $\mathbf{TxtMex}(h, \psi) = \{\mathcal{L} \mid (\exists \mathrm{M})[\mathcal{L} \subseteq \mathbf{TxtMex}(\mathrm{M}, h, \psi)]\}$.
(c) $\mathbf{TxtMex}(\psi) = \{\mathcal{L} \mid (\exists h \in \mathcal{R})[\mathcal{L} \in \mathbf{TxtMex}(h, \psi)]\}$.

It is easy to see that for all $h$, $\mathbf{TxtMex}(h, \psi) \subseteq \mathbf{TxtMex}(\lambda x.[h(x) + 1])$. The following theorems show that there exist acceptable programming systems for which the above inclusion is (is not) proper.

**Theorem 2** *Let $h_0, h_1, h_2, \ldots$ be an infinite r.e. sequence of distinct non-decreasing recursive functions such that $(\forall x)[h_i(x) > x]$.* $(\exists \psi)$ $(\forall i)[\mathbf{TxtMex}(h_i, \psi) \supset \mathbf{TxtMex}(\lambda x.[h_i(x) - 1], \psi)]$.

**Theorem 3** *Let $h_0, h_1, h_2, \ldots$ be an infinite r.e. sequence of distinct non-decreasing recursive functions such that $(\forall x)[h_i(x) \geq x]$.* $(\exists \psi)(\forall i)[\mathbf{TxtMex}(h_i, \psi) = \mathbf{TxtMex}(\lambda x.[x], \psi)]$.

Kinber [Ki83], in the context of function inference, considered a generalization of minimal-identification. He showed some initial results about a learning criterion in which, for some positive integer $i$, an inductive inference machine, when fed the graph of a recursive function, is required to converge to the $i^{th}$ minimal program in the acceptable programming system $\psi$. We study an even more general learning criteria.

**Definition 5** Suppose $L \in \mathcal{E}$ and $i \in N^+$. We say that $k$ is the $i^{th}$ $\psi$-grammar for $L$ (written: $k = i\text{-}\mathbf{mingrammar}_\psi(L)$) $\iff [[W_k^\psi = L] \wedge [\|\{j \mid (j < k) \wedge (W_j^\psi = L)\}\| = i - 1]]$.

Definition 6 below describes our generalized minimal identification criteria in the context of language learning.

**Definition 6** Suppose $h \in \mathcal{R}$ be such that for all $x$, $h(x) > 0$.
(a) **M** $h$-$\mathbf{TxtMin}_\psi$-*identifies* $L$ (written: $L \in h\text{-}\mathbf{TxtMin}_\psi(\mathbf{M})$) $\iff$ ($\forall$ texts $T$ for $L$)($\forall^\infty n$) $[\mathrm{M}(T[n]) = h(\mathbf{mingrammar}_\psi(L))\text{-}\mathbf{mingrammar}_\psi(L)]$.
(b) $h\text{-}\mathbf{TxtMin}_\psi = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathrm{M})[\mathcal{L} \subseteq h\text{-}\mathbf{TxtMin}_\psi(\mathrm{M})]\}$.

Clearly, $(\lambda x.[1])$-$\mathbf{TxtMin}_\psi$-identification is the same as $\mathbf{TxtMin}_\psi$-identification.

**Theorem 4** $(\forall \psi)(\forall$ *non-decreasing* $h_1, h_2 \in \mathcal{R}$ *such that forall* $x$, $h_1(x) > 0 \wedge h_2(x) > 0)[(\forall x)[h_1(x) \geq h_2(x)] \Rightarrow [h_1\text{-}\mathbf{TxtMin}_\psi \subseteq h_2\text{-}\mathbf{TxtMin}_\psi]]$.

**Theorem 5** *Let* $h_0, h_1, h_2, \ldots$ *be an infinite r.e. sequence of distinct recursive functions* $\in \mathcal{R}^+$. $(\exists \psi)(\forall i)[h_i\text{-}\mathbf{TxtMin}_\psi \supset (\lambda x.[h_i(x) + 1])\text{-}\mathbf{TxtMin}_\psi]$.

**Corollary 1** [Ki83] $(\exists \psi)(\forall c > 0)$ $[(\lambda x.[c])\text{-}\mathbf{TxtMin}_\psi \supset (\lambda x.[c + 1])\text{-}\mathbf{TxtMin}_\psi]$.

**Theorem 6** *Let* $h_0, h_1, h_2, \ldots$ *be an infinite r.e. sequence of distinct recursive functions* $\in \mathcal{R}^+$. $(\exists \psi)(\forall i, j)$ $[h_i\text{-}\mathbf{TxtMin}_\psi = h_j\text{-}\mathbf{TxtMin}_\psi]$.

# 5 Discussion

In the previous section, we presented some results that show the dependence of learning criteria resulting from the requirement that machines converge to 'small' size grammars. We are able to show similar results for a number of other formulations of succinctness. On first observation, these results seem to say that language learning criteria resulting from seemingly 'natural' notions of succinctness are uninteresting (or, mathematically dirty) as they are dependent on something as insignificant as the names of programs. However, we are also able to show that some of these dependence results still hold if we restrict our attention to a very 'nice' subclass of programming systems called Kolmogrov numberings (These programming systems are in some sense 'the most informative' ones, as by definition, every acceptable programming system can be reduced to a Kolmogrov numbering via a recursive function with no more rapid than linear growth (Freivalds [Fr90])).

These results seem to suggest that complexity restrictions on general models of language acquisition will most likely result in learning criteria which are dependent on the choice of acceptable programming system. This dependence may be a very fundamental fact about language acquisition rather than a mathematical inconvenience.

For further results and proofs of the theorems see [JS90d].

## 6 Acknowledgements

We would like to express our gratitude to John Case and Mark Fulk for advice and encouragement. Our results build on techniques of Freivalds, Kinber, and Chen. Work was done while the first author was at the University of Rochester and the second author was at the University of Delaware. Sanjay Jain was supported by NSF grant CCR 832-0136 at the University of Rochester and Arun Sharma was supported by NSF grant CCR 871-3846 at the University of Delaware and by a grant from the Siemen's Corporation at MIT.

## 7 Bibliography

[Bl67a] Blum, M. *On the size of Machines.* Information and Computation, Vol. 11, 1967. Pages 257-265.

[Bl67b] Blum, M. *A Machine Independent Theory of the Complexity of Recursive Functions.* Journal of the ACM, Vol. 14, 1967. Pages 322-336.

[CC86] Case, J. and Chi, *Machine Learning of Nearly Minimal Size Grammars.* Unpublished Manuscript.

[Ch82] Chen, K. *Tradeoffs in Inductive Inference of Nearly Minimal Sized Programs*, Information and Control, Vol. 52, 1982. Pages 68-86.

[Fr75] Freivalds, R. *Minimal Gödel Numbers and Their Identification in the Limit*, Lecture Notes in Computer Science, Vol. 32, 1975. Pages 219-225.

[Fr90] Freivalds, R. *Inductive Inference of Recursive Functions: Qualitative Theory*, Unpublished.

[Go67] Gold, E. M. *Language Identification in the Limit*, Information and Control, Vol. 10, 1967. Pages 447-474.

[JS90d] Jain, S and Sharma, A. *Program Size Restrictions in Inductive Learning.* Technical Report no. 90-06, Department of Computer and Information Sciences, Univ. of Delaware.

[Ki83] Kinber, E. B. *A note on the Limit Identification of c-minimal Indices.* EIK Vol. 19, 1983. Pages 459-463.

[MY78] Machtey, M. and Young, P. *An Introduction to the General Theory of Algorithms*, North Holland, New York, 1978.

[OSW86] Osherson, D., Stob, M. and Weinstein, S. *Systems that Learn, An Introduction to Learning Theory for Cognitive and Computer Scientists*, MIT Press, Cambridge, 1986.

[Pn79] Pinker, S. *Formal Models of Language Learning*, Cognition, Vol. 7, 1979. Pages 217-283.

[Ro58] Rogers, H. *Gödel Numberings of Partial Recursive Functions*, Journal of Symbolic Logic, Vol. 23, 1958. Pages 331-341.

[Ro67] Rogers, H. *Theory of Recursive Functions and Effective Computability*, McGraw Hill, New York, 1967.

[WC80] Wexler, K. and Culicover, P. *Formal Principles of Language Acquisition*, MIT Press, Cambridge, 1980.

[We82] Wexler, K. *On Extensional Learnability*, Cognition, Vol 11, 1982. Pages 89-95.

# Acquiring the Semantics of Simple Phrasal Patterns Using COBUILD

Narciso Jaramillo and Marti Hearst*

Computer Science Division, 571 Evans Hall
University of California, Berkeley
Berkeley, CA 94720
*nj@teak.Berkeley.EDU, marti@teak.Berkeley.EDU*

## Abstract

Many researchers are studying the possibility of acquiring the syntax and semantics of lexical items from machine readable dictionaries, and others are exploring the application of statistical metrics to large text corpora in order to uncover useful correlations among lexical items. We are interested in using both machine readable dictionaries and correlation statistics from large corpora to aid in the acquisition of the semantics associated with certain kinds of lexical patterns. In this paper we only briefly outline how to tie the two techniques together, focusing mainly on lexical acquisition from a dictionary (specifically the *Collins COBUILD English Language Dictionary*). We describe some methods for determining the semantics of simple patterns consisting of a noun or verb followed by a prepositional phrase, and compare this method to existing techniques.

## Biographies

**Narciso Jaramillo** and **Marti Hearst** are Ph.D. students in the Berkeley Artificial Intelligence Research group (BAIR) at the University of California, Berkeley, working with advisor Robert Wilensky.

Jaramillo is studying the feasibility of extracting useful semantic information from definitions in the *Collins COBUILD English Language Dictionary.*

Hearst is exploring coarse-grained approaches to text interpretation and lexical acquisition. She participated in the AAAI Spring Symposium on Text Based Intelligent Systems (1990), describing a method for placing structure on a corpus by sorting the documents into categories based on limited semantic and syntactic analysis (Hearst 1990). More recently, she has been developing an accurate, relatively low-overhead method for the disambiguation of English noun homonyms using a large corpus of free text (Hearst 1991).

## References

Hearst, M. A. (1990). A hybrid approach to restricted text interpretation. In P. S. Jacobs, editor, *Text-Based Intelligent Systems: Current Research in Text Analysis, Information Extraction, and Retrieval*, pages 38–43. GE Research & Development Center, TR 90CRD198.

Hearst, M. A. (1991). Toward noun homonym disambiguation using local context in large text corpora. Submitted to *The Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics.*

*This material is based on work supported by a grant from the Department of Defense Advanced Research Projects Agency, monitored by the Office of Naval Research under grant number N00014-89-J-3205, as well as by a National Science Foundation Graduate Fellowship awarded to the first author.

# 1 Introduction

Automatic language acquisition is a promising approach for the creation and augmentation of lexicons for natural language processing applications. Currently many researchers (e.g. (Wilks *et al.* 1991), (Boguraev & Briscoe 1987), (Ahlswede & Evens 1988), (Jensen & Binot 1987)) are exploring the transformation of machine readable dictionaries into what Wilks *et al.* calls "machine **tractable** dictionaries", i.e. dictionaries transformed into a format usable for NLP. Most of this work focuses on obtaining word senses or word "relatedness" information.

Researchers are also exploring the application of statistical metrics to large text corpora in order to detect patterns such as noun similarity (Hindle 1990), collocation occurrence (Choueka 1988), (Smadja & McKeown 1990), and verb alternations (Brent 1990). These approaches uncover correlations among lexical items that are potentially useful for more semantically driven acquisition tasks.

We are interested in using both machine readable dictionaries (specifically, *The Collins COBUILD English Language Dictionary* (Sinclair 1987)) and correlation statistics from large corpora to aid in the acquisition of the semantics associated with certain kinds of lexical patterns. In particular, the work we discuss in this paper focuses on acquiring semantic representations of patterns composed of a noun or a verb followed by a prepositional phrase. In many cases, the meanings of these patterns are not simple compositions of the meanings of their parts, so prior knowledge about the meanings of the individual words may be inadequate for the interpretation of the phrase.

Section 2 presents an overview of the acquisition method, Section 3 describes application of the method to several examples, Section 4 provides a comparison between this method and previous approaches, and Section 5 concludes the paper.

# 2 Overview of the Acquisition Algorithm

## 2.1 Methods

From a corpus, we selected several dozen sentences containing noun + preposition and verb + preposition patterns. We found that three methods relying on information from COBUILD were sufficient for creating a representation of their meanings. Briefly, these methods are:

(1) **Interpret Definition Directly:** Some noun/verb + preposition patterns are directly defined in the dictionary—that is, the noun + preposition pair or verb + preposition pair is used in the definition of the main noun or verb. For example, the definition for fluent 1 begins, "Someone who is **fluent** in a particular language, or who speaks **fluent** Spanish, French, Russian, etc . . . ," indicating that the patterns "fluent in *language*" and "speaks fluent *language*" are defined therein. In cases like these, the definition is parsed to find semantic constraints on complement structure of the pair. This in turn can require analysis of the definitions of other words or phrases, a mechanism we label "definition hopping."

(2) **Use Knowledge about How Nouns Derive from Verbs:** For noun + preposition pairs in which the noun is derived from a verb, we can apply some standard rules to derive the noun's complementation structure from that of the verb. This is especially useful because often only the verb + preposition combination appears explicitly in the dictionary.

(3) **Use Hand-coded Prepositional Semantics:** This is the catch-all category. In many cases COBUILD provides an appropriate definition for the preposition that indicates the relationship between the noun or verb and the preposition's object. These definitions often describe the preposition's function rather than defining it in an ordinary fashion, and so may not be automatically acquirable. However, since COBUILD provides a fairly thorough analysis of prepositional meanings, we can use their definitions as a starting point for hand-coding their semantics.

We chose to study COBUILD in our research for several reasons. Unlike most dictionaries, COBUILD's definitions are written in complete sentences. For example, COBUILD's definition of **decay** (omitting nominal senses) is as follows:

> 1 When something such as a plant, a piece of wood, or a piece of meat **decays**, it becomes rotten or unusable.
>
> 2 If something such as a social or political institution **decays**, it gradually becomes weaker or more corrupt.

This presentation style, among other reasons, makes this dictionary amenable to general-purpose parsing. Furthermore, it provides us with information about syntactic and semantic features of typical complements of words. When we encounter a polysemous word used in conjunction with a particular complement, we can determine how well that complement fits the description given in the definition. To make this determination, we can use several kinds of background knowledge, the simplest of which is synonymy/hypernymy information extracted from the dictionary itself. Such information can be gathered from a genus hierarchy or from COBUILD's margin annotations. Having these relationships, especially the hypernymy relationship, indicated explicitly may make the task of creating a semantic hierarchy easier than when using other dictionaries (see (Guthrie *et al.* 1990) for a discussion of some of the difficulties associated with this task). Additionally, if more refined knowledge from

other sources exists in the knowledge base, that can be used to aid in determining whether the complement used fits the description.

If the complement preferences are insufficient to uniquely determine the sense of the word in its given usage, we can use some simple heuristics to choose between the possibilities:

**Specificity:** In general, of a series of senses whose complement preferences match the current context of the word in question, we choose the sense with the most specific preferences, since it is likely to provide the most accurate information about the given usage.

**Definition order:** Different dictionaries order their definitions differently. According to the introduction to COBUILD, the first sense given for a particular headword is not necessarily the most common sense; it may be the sense which the lexicographer felt was the most "central" in meaning. But because of this, earlier definitions are likely to be more general, and therefore be at least coarsely appropriate for the given usage.

These two heuristics may seem to be at odds with each other—specificity would tend to favor more informative definitions, while earlier definitions are more general. Specificity is useful in situations in which several compatible definitions have fairly restrictive complement preferences; definition order is a more conservative "fallback" heuristic, more useful in situations in which few or no senses with nontrivial complement preferences exist. Therefore, given a conflict between the two heuristics, we prefer specificity over definition order.

Though these heuristics are not in general adequate for disambiguation of ordinary text, we assume that they will be useful in the restricted domain of dictionary definitions. Whether this assumption is justified remains to be empirically verified.

## 2.2 Resources

In order to make use of dictionary definitions at all, we will need to hand-code the semantics of a basic set of words. While COBUILD does not have an explicit "defining vocabulary" as does LDOCE (Summers 1987), another learner's dictionary,[1] we can use methods similar to those of Guo in (Wilks *et al.* 1991) to determine a core set of defining terms.[2] Currently we have hand-coded just enough basic words to interpret the definitions we have been working on; we will build a larger set of basic words in time.

To parse COBUILD's definitions, we plan to use a Construction Grammar-based parser (Jurafsky 1990) when it becomes available. In the Construction Grammar formalism (Fillmore 1989), each grammatical element (e.g. phrases, clauses, sentences) is viewed as the unification of a collection of constructions that indicate syntactic, semantic, and pragmatic information simultaneously. Our initial representation language and inference mechanism is a recent version of KODIAK (Wilensky 1986), although it would be propitious to extend it to accommodate phrases using some of the techniques described in (Besemer & Jacobs 1987).

We plan to use the corpus to select noun/verb + preposition pairs to be learned, using methods that compute statistics over a large text corpus, such as those described in (Smadja & McKeown 1990). This is an alternative to having the system process pairs on an "as-needed" basis, with a text-understanding system initiating the acquisition process when it finds an unknown pair or unknown usage of a known pair. It may be desirable to have statistical evidence that a particular usage of a pair is frequent enough to merit entering it in the lexicon. Furthermore, by examining the contexts in which a pair occurs, it may be possible to determine whether or not it has a non-compositional meaning and so requires detailed semantic analysis. Another potential use of the corpus is to provide extra information when the dictionary entry is not detailed enough for the semantic analysis. We have not explored these last two options in detail; the remainder of this paper concentrates on a description of our use of COBUILD for semantic acquisition.

# 3 Examples

In this section, we present examples of how the dictionary, combined with the resources mentioned above, can be used to interpret the meanings of patterns consisting of a noun or a verb followed by a prepositional phrase. The following subsections give some examples of patterns that can be analyzed using each of the three methods described in Section 2.

## 3.1 Direct definition

Suppose we want to understand the meaning of **condescend to** in the following sentence:[3]

> Harris always condescended to waiters and servants, making snide remarks about their station in life.

In COBUILD, we find the following definitions for **condescend**:

> 1 [V: IF +PREP THEN *to*] If you **condescend** to people, you behave in a way which shows

[1] We are preparing a report which compares COBUILD with LDOCE as sources for semantic acquisition.

[2] Part of this process consists of finding "cliques" of definitions, all of which refer only to one another, labeling these groups as primitives and hand coding them. This phenomenon of circular definitions-as-primitives is also noted in (Amsler 1981) and (ichi Nakamura & Nagao 1988).

[3] This sentence was created for illustrative purposes; it was not derived from a corpus, but the methods described herein were developed using naturally occurring sentences.

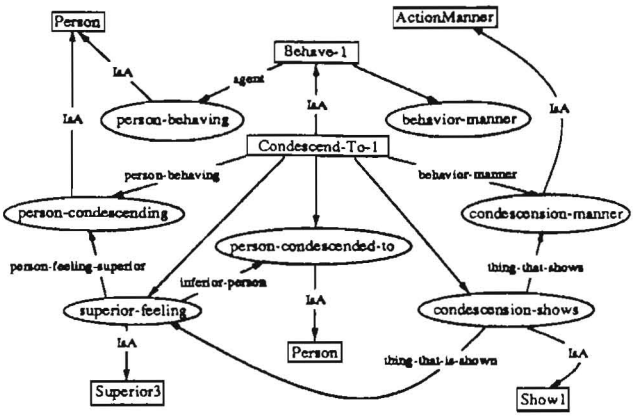



Figure 1: Interpretation of **condescend 1**

them that you think that you are superior to them.

**2** [V+*to*-INF / = deign] If you **condescend** to do something, you agree to do it, but in a way that shows that you think that you are doing people a favor; used showing disapproval.

We can immediately eliminate sense 2 by syntactic restrictions, since the object of **to** in the sentence is a noun, not an infinitive verb. The first clause of the definition informs us that the subject of **condescend** and the object of **to** must be human beings, a fact we can easily verify from knowledge about Harris (presumably from interpreting earlier parts of the text), waiters and servants. Thus, it seems likely that sense 1 is the proper sense in this context.

The syntactic complement structure of **condescend 1** is easily determined from its definition above. Our interpretation of the semantics of the definition is shown in Figure 1. Briefly, rectangles contain the names of relations; ovals contain the names of participants in those relations. Unlabelled links point to participants specific to a particular relation, while labelled links (except for IsA) point to roles inherited from a parent. For example, the person-behaving link between Condescend-To-1 and person-condescending indicates that the person-condescending plays the role in the Condescend-To-1 relation that person-behaving plays in the Behave-1 relation.[4]

Let us examine how our algorithm produces this representation. First, we know from the definition that there are two semantic participants directly realized as syntactic complements of **condescend 1**; in the diagram, these are represented by person-condescending (realized as the subject of **condescend**) and person-condescended-to (realized as the object of **to**). The other participants are implied by the rest of the definition, whose interpretation we now describe. For this example, we will assume that appropriate senses of **way**, **show**, and **think** are available in the basic hand-coded vocabulary. Given these, we still need to find interpretations for **behave** and **superior** that are syntactically and semantically compatible with their usages in the definition of **condescend 1**. Of the senses of **behave** defined in COBUILD, only two are syntactically compatible:

**1** If you **behave** in a particular way, you act in this way, especially because of the situation you are in or the people you are with.

**3** If an object, substance, etc **behaves** in a particular way, it functions in a way that follows the laws of science.

The complement preferences given for the subject of each sense of **behave** here do not unequivocally disambiguate the use at hand; the fact that the subject of **condescend 1** is a person is not incompatible with its being an object, since people are objects. Thus, we have to make a decision between the two. Both of the heuristics mentioned in Section 2, specificity and definition order, would lead us to choose **behave 1** as the proper sense in this context.

We will not go into detail about the interpretation of the definition of **behave 1**; let us assume that we have interpreted it and stored its syntax and semantics in its lexical entry, calling its associated concept Behave-1, and that it has (at least) the two roles shown in the diagram. We can now establish that Condescend-To-1 IsA Behave-1. We know that the subject of **condescend** is also the subject of behave; thus, the person-condescending participant in the Condescend-To-1 relation plays the role of the person-behaving participant in the Behave-1 relation. We create the condescension-manner participant and the behavior-manner link similarly.

We now turn our attention to **superior**. There are three senses in COBUILD that are syntactically compatible with its use in **condescend 1**. Of these, the most appropriate in this context is **superior 3**:

**3** If you feel **superior** to other people, you believe that you are better than they are. You often make people aware of your attitude by your expression or tone of voice or by the way you treat them.

However, in order to realize that this sense applies, we must assume that **think** [**that**] and **feel** are similar in meaning. In this case, the dictionary can help us: The definitions of the appropriate senses of each of these have annotations in the margin indicating that they are both hyponyms of **believe**. **Superior 3** has more restrictive complement preferences than the other two senses, and

[4] The names given to the relations and participants here are provided for clarity; they should not be taken as having any semantic content in and of themselves.

it is compatible with the usage of **superior** in **condescend 1**; therefore, we would choose it over the other two senses.

After studying whatever unknown words are used in the definition of **superior 3**, we can construct its syntactic and semantic representation (not shown in the diagram). We then create the participant superior-feeling to our representation of Condescend-To-1, noting that it is an instance of Superior-3. From the definition of **condescend 1**, we understand that the person who feels superior is the person-condescending, and the person considered inferior is the person-condescended-to; we can therefore add these links to the representation. The information about the condescension-shows participant can be similarly derived from the definition.

Note that there is nothing corresponding to **think that** in the structure. Since we concluded that **think that** is equivalent to **feel**, and since **feel** is included in the complement preferences of **superior 3**, we can assume that the definition of **superior 3** must include whatever semantics can be attributed to **feel** (as indeed it does). Therefore, we can allow the concept associated with **superior 3** to take care of the semantics of **think that/feel.**

### 3.2 Nouns derived from verbs

Many nouns which take prepositional complements are derived from verbs, and we can exploit some regularities in English to understand these noun + preposition patterns. In general, they fall into two categories:

**TransVerb → Noun + PP[*of*]** Nominal forms of transitive verbs often take prepositional phrases headed by **of** as complements, where the object of **of** plays a similar semantic role to the object of the original verb. For example, the use of **separation** in "the separation of the executive and judicial branches [by the Constitution]" is nominally related to the transitive verb **separate**; the phrase is semantically similar to "[The Constitution] separated the executive and judicial branches."

**Verb + PP[x] → Noun + PP[x]** In cases where the original verb takes a particular prepositional complement, its derived noun often takes the same prepositional complement. For example, in "Provision was made for the project's eventual termination," the **PP[*for*]** complement of **provision** springs from the **PP[*for*]** complement of **provide**, as in "They provided for the project's eventual termination."

Thus, even if a particular noun + preposition pattern is not directly defined in the dictionary, we can use the dictionary or morphological analysis to determine whether the noun is derived from a verb, and, if so, interpret the role of the preposition's object accordingly. (Of course, if the noun is not directly defined in the dictionary, we will have to make recourse to semantic regularities in order to derive the meaning of the noun from the meaning of the original verb.)

### 3.3 Prepositional semantics

If the noun/verb + preposition pattern is not explicitly mentioned in the dictionary at all, and neither of the regularities in the last section are applicable, we must attempt to integrate knowledge about the individual words in the pattern in a plausible way. We might begin by applying our standard dictionary-interpretation techniques to derive the semantics of the main noun or verb and the preposition. However, prepositions often have complex semantics; the dictionary does not always give complete definitions for prepositional senses.

For example, suppose we wish to interpret the phrase "jar of mayonnaise." The appropriate sense in COBUILD of **of** for this phrase is **1.1**: "You use **of** after nouns expressing quantities, groups, measurements, amounts." However, the definition COBUILD gives for the noun **jar** is "A **jar** is a container ... that has a wide top and is used for storing food ..." In order to obtain the proper sense of **of**, we must realize that containers can be schematized as units of measurement; this would require knowledge about conventional uses of containers. However, even given that we can identify this as the proper sense of **of**, its definition is incomplete, since it does not specify precisely what the relation is between "jar" and "mayonnaise".

Thus, it seems reasonable to include prepositions in the basic vocabulary, and code their semantics by hand. We can use COBUILD's definitions as a starting point, using the dictionary's analysis of the various prepositional senses, but making their meanings explicit in our knowledge representation language. As before, we can use the individual senses' complement preferences as disambiguational clues. We have only studied a few prepositions in detail, but COBUILD's sense analysis and complement preferences seem to account for a significant percentage of cases not covered by the other two methods.

## 4 Comparison to Other Methods

The way in which definitions are analyzed here is in some ways similar to other approaches that extract semantics from dictionary definitions. For example, several approaches (e.g. (Amsler 1981),(Wilks *et al.* 1991), (Jensen & Binot 1987)) make use of taxonomic relations (hypernyms) for creating a semantic hierarchy, which aids in "definition hopping" (although as mentioned above, this task is made easier by COBUILD's margin annotations). Approaches such as (Alshawi 1987),

(Boguraev & Briscoe 1987), and (Wilks *et al.* 1991) assume that a set of 1000 - 3000 primitive senses need to be pre-encoded in the lexicon, as does our algorithm. (Jensen & Binot 1987) describes a detailed encoding of the semantics of the preposition *with*; the approach described here requires similar prepositional encodings, but interprets them using more generally applicable mechanisms.

A significant way in which this approach differs from others is linked to the COBUILD definition style. As noted above, COBUILD's definitions are written as complete and fairly simple sentences, and so can be parsed and interpreted with a general-purpose parser and grammar, which can be used in other tasks. In contrast, most approaches interpret the definitions by matching them against specially-tailored patterns (e.g. (Alshawi 1987),(Markowitz *et al.* 1986), (ichi Nakamura & Nagao 1988)), or by using special-purpose parsers (e.g. (Wilks *et al.* 1991), (Zernik & Dyer 1985)) in order to extract particular semantic relations. This is done both because the definitions of the dictionary used are more terse than ordinary language, and so cannot be properly parsed, and because the dictionaries have enough uniformity in format to allow this to work successfully. Ahlswede & Evens (1988) presents an interesting comparison between using a general parser and coarse text-processing tools for the derivation of semantic relations from a dictionary. They concluded that for the semantic relations they were acquiring, the text processing tools were more appropriate than the general purpose parser. However, they also noted that parsing would have been useful for verb definitions whose headword is a verb plus a particle.

# 5 Conclusion

As we have shown, dictionaries can facilitate the interpretation of patterns larger than a single word, either by directly defining them, or by providing information that can be used in combination with other knowledge about language and the world. Statistical analysis of text corpora may prove useful in determining what kinds of patterns to interpret and in providing appropriate context in which to interpret these patterns. The interpretation of dictionary definitions, when combined with some basic knowledge resources, can provide broad support for extended knowledge acquisition and natural language processing tasks. Other acquisitional tools can then refine the knowledge we obtain from the dictionary by consulting other, more detailed information sources.

# References


Ahlswede, T. & M. Evens (1988). Parsing vs. text processing in the analysis of dictionary definitions. *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, pages 217–224.

Alshawi, H. (1987). Processing dictionary definitions with phrasal pattern hierarchies. *American Journal of Computational Linguistics*, 13(3):195–202.

Amsler, R. A. (1981). A taxonomy for english nouns and verbs. *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics*, pages 133–138.

Besemer, D. J. & P. S. Jacobs (1987). Flush: A flexible lexicon design. *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 186–192.

Boguraev, B. & T. Briscoe (1987). Large lexicons for natural language processing: Utilising the grammar coding system of ldoce. *American Journal of Computational Linguistics*, 13(3):203–218.

Brent, M. (1990). Semantic classification of verbs from their syntactic contexts: Automated lexicography with implications for child language acquisition. *COGSCI90*, pages 428–437.

Choueka, Y. (1988). Looking for needles in a haystack, or locating interesting collocational expressions in large textual databases. *Proceedings of the RIAO*, pages 609–623.

Fillmore, C. (1989). The mechanisms of "construction grammar". In *Proceedings of the Berkeley Linguistic Society*, volume 15.

Guthrie, L., B. M. Slator, Y. Wilks, & R. Bruce (1990). Is there content in empty head? In *Proceedings of the Thirteenth International Conference on Computational Linguistics*, Helsinki.

Hindle, D. (1990). Noun classification from predicate-argument structures. *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 268–275.

ichi Nakamura, J. & M. Nagao (1988). Extraction of semantic information from an ordinary english dictionary and its evaluation. In *Proceedings of the Twelfth International Conference on Computational Linguistics*, pages 459–464, Budapest.

Jensen, K. & J.-L. Binot (1987). Disambiguating prepositional phrase attachments by using on-line dictionary definitions. *American Journal of Computational Linguistics*, 13(3):251–260.

Jurafsky, D. (1990). Representing and integrating linguistic knowledge. In *Proceedings of the Thirteenth International Conference on Computational Linguistics*, Helsinki.

Markowitz, J., T. Ahlswede, & M. Evens (1986). Semantically significant patterns in dictionary definitions. *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pages 112–119.

Sinclair, J., editor (1987). *The Collins COBUILD English Language Dictionary*. William Collins Sons & Co Ltd, Glasgow.

Smadja, F. A. & K. R. McKeown (1990). Automatically extracting and representing collocations for language generation. *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 252–259.

Summers, D., editor (1987). *The Longman Dictionary of Comtemporary English*. Longman Group UK Limited, Harlow.

Wilensky, R. (1986). Some problems and proposals for knowledge representation. In J. L. Kolodner & C. K. Riebeck, editors, *Experience, Memory, and Reasoning*, pages 15–28. Lawrence Erlbaum, New Jersey.

Wilks, Y. A., D. C. Fass, C. ming Guo, J. E. McDonald, T. Plate, & B. M. Slator (1991). Providing machine tractable dictionary tools. To Appear in *Computers and Translation*.

Zernik, U. & M. Dyer (1985). Towards a self-extending lexicon. *Proceedings of the 23th Annual Meeting of the Association for Computational Linguistics*, pages 284–292.

# On Building a Model of Grammar from Information in the Lexicon

Rick Kazman
Department of Philosophy and
Software Engineering Institute
Carnegie Mellon University

Abstract

Much of the syntax of the world's languages may be characterized by the inventory and properties of the lexical items and *Functional Categories* (FCs) of those languages. FCs are the "little words" of a language: determiners, auxiliaries, complementizers, prepositions and (inflectional) affixes.

In this paper, I will be investigating the proposal that syntax is acquired by the child as a progression from an invariant base (a core grammar which is common to all languages) to a more articulated view of language which includes FCs. Given, however, that the inventory of FCs differs from language to language, an explicit proposal of how children acquiring a language come to learn its FCs is needed.

I propose that the FCs of a language are originally lacking in the child's syntax, and are acquired by the child through an analysis of the agreement facts of the language. This information is available to the child through a lexical acquisition process which distinguishes the invariant part of a word (the root) from the part which varies according to changes in salient features of those words (the affixes). This procedure is intended to be psychologically plausible—it is sensitive to the frequency, phonological and semantic salience of words in the input, and makes predictions about order of acquisition and overgeneralizations which are corroborated by studies of lexical acquisition and psychological studies on the nature of the mental lexicon. By examining the generalizations made by the lexical acquisition procedure, the child has a sufficiently broad understanding of the agreement processes in his language to be able to hypothesize the FCs of his language.

This proposal makes several predictions for the time course of acquisition: that lexical information about a category will be acquired in direct relation to the frequency and salience of that category in the input language; that FCs will originally be missing from the child's grammar; that the agreement information for a category (the affixes) will be acquired before the FCs for that category. These predictions have been shown to hold for English, French, Polish, Dutch and Hebrew.

When seen this way, the acquisition of syntax is not simply a logical problem for the child to solve, as has been advocated in the past, it is a statistical inductive procedure which the child can only solve after exposure to significant amounts of data.

## Research interests

My interests tend to center in three broad largely unrelated areas: human-computer interfaces, the creation and manipulation of large text/natural language databases and computational models of language acquisition. I'll make the assumption, and I think it's a safe one, that the first two of my interests are of little interest to workshop members, and I'll focus on the third.

Within computational models of language acquisition, I have largely been concerned with creating psychologically and computationally valid models of the acquisition of the lexicon and syntax. I have concentrated mainly on the early stages of acquisition, from about 18 to 36 months, when the basic grammar of a language is acquired.

My intent is to create models which both account for the longitudinal facts of acquisition across languages. These models should, furthermore, have something to say about the adult state. That is, it should be clear how the child model of language becomes the adult model, and this transformation should not involve the introduction of any additional principles. In particular, I have been concentrating on the connection between morphology and syntax, and looking at how the child might use information contained in the lexicon to deduce facts about the syntax of his language.

## Selected Bibliography

Rick Kazman, (1988). *Null subjects and the acquisition of Case and Infl.* Presented to the 13th Boston University Conference on Language Development.

Rick Kazman, (1990). *The induction of the lexicon and the early stages of grammar.* Unpublished ms., Carnegie Mellon University.

Rick Kazman, (1990). *The Genesis of Functional Categories.* Presented to the 15th Boston University Conference on Language Development.

# 1 Introduction

In recent years linguistic theories have come to rely more and more heavily on a set of linguistic universals, and on knowledge contained in the lexicon, and less heavily on rules which are specific to a particular grammatical construction (see, for example Chomsky's *Lectures on Government and Binding* [Cho81], and and Pollard and Sag's *An Information-based Syntax and Semantics* [PS87]. That is, the structure of a natural language can largely be specified by the properties of individual lexical items, and by the properties of lexical categories, rather than by a set of rules which detail the phrase structure of the language. This has intuitive appeal in that, seen this way, languages are of a single basic form, paralleling hypothesized innate cognitive abilities. In a theory of grammar these abilities are expressed as structural limitations on the possible form of human natural languages. The differences between languages are encoded in the properties of the lexical items and categories of each language.

Until now, however, this characterization of natural languages has not been extensively tested as a theory of language acquisition. I am proposing a model of language acquisition which builds a language representation from a simple common base, tailoring the representation based upon information contained in the lexicon. This tailoring is manifested as the construction of the language's *functional categories* (FCs) [Abn87].[1] FCs, as a class, are distinguished from the class of major *thematic categories* (TCs) of a language (verbs, nouns, adjectives and adverbs). Many language acquisition researchers have noted that FCs are acquired as a group by the child ([GN88], [Kaz88]). To account for these facts, I propose a model of lexical and syntactic acquisition which explicitly links the mastery of agreement paradigms to the creation of FCs.

# 2 The Lexical Acquisition Model

The model of lexical acquisition which I will propose is intended to be a computationally, psychologically and empirically accurate reflection of the child's acquisition of the lexicon, with respect to the period of language acquisition paralleling that of an 18 to 30 month old child. During this period, most of the basic vocabulary and grammar of the target language are acquired. Furthermore, the lexical acquisition model will be related to a model of syntactic acquisition. I will show how the results of the morphological analysis provide just the information needed by the syntactic acquisition procedure in order to correctly characterize agreement relations in the target language.

This work was originally motivated by the desire to provide a principled account of certain observations noted by language acquisition researchers:

1. Children morphologically overgeneralize.
2. The rate of acquisition of lexical items is directly proportionate to the input frequency and phonological salience of these items.
3. Agreement paradigms are acquired non-monotonically: children's mastery of particular agreement paradigms does not steadily increase, but appears to vary, even from utterance to utterance.
4. Function words, case-marking and syntactic agreement are acquired as a group.

The lexical acquisition procedure accepts parsed sentences as input, one word at a time, and creates lexical entries to represent the input words. The procedure attempts to distill the core meaning of a word by comparing its use across many sentences. The parser used is simple and general—it assumes that all features are shared by all words within a phrase and that a phrase can attach to another phrase either as an argument or as a modifier. Conceptually, the acquisition procedure is composed of two functions: 1) a version space procedure [Mit77] which compares the features of words in the input in order to arrive at the minimal set of defining features for each particular word; and 2) a word segmentation procedure which compares different versions of the same word (in the example below, the words *bite* and *bites* are compared) in order to isolate a root—which contains the core meaning and features of the word—and a number of affixes, which annotate the features of the root (the *-ed* affix, for example, add the past-tense feature TNS=PAST to the root).

As a concrete example, consider input sentence 1 (given in its parsed, fully annotated, format):

```
1. The dogs bite the cats.
```

[1] Examples of FCs in English are Complementizer (that, for, which, etc.), Inflection (can, to, will, the past tense affix *-ed* etc.) and Determiner (the, a, my, etc.).

```
[WORD:[dh ah0] POS:det SEM:the
IFEATS:[TH=AGT PER=3 NUM=PL DEF=Y] EFEATS:[ROLE=SUB] ARGS:[[SUB=NOUN]] ]

[WORD:[d ao1 g z] POS:noun SEM:dog
IFEATS:[TH=AGT PER=3 NUM=PL DEF=Y] EFEATS:[ROLE=SUB] ]

[WORD:[b aa1 iy t] POS:verb SEM:bite
ARGS:[ [TH=AGT PER=3 NUM=PL DEF=Y ROLE=SUB] [TH=PAT PER=3 NUM=PL DEF=Y ROLE=OBJ]]
IFEATS:[TNS=PRS MOOD=IND] ]

[WORD:[dh ah0] POS:det SEM:the
IFEATS:[TH=PAT PER=3 NUM=SG DEF=Y] EFEATS:[ROLE=OBJ] ARGS:[[SUB=NOUN]] ]

[WORD:[k ae1 t] POS:noun SEM:cat
IFEATS:[TH=PAT PER=3 NUM=SG DEF=Y] EFEATS:[ROLE=OBJ] ]
```

Each word in the input presented to the lexical acquisition procedure has the following properties: a phonetic string (e.g. cats = k ae1 t s),[2] a part of speech, a semantic identification token (typically the word itself is used, but any unique identifier would do), a list of internal features, external features and (optionally) arguments. Consider the two lexical entries for the determiner *the* above. The lexical acquisition procedure will compare the two instances of *the*, and will produce the generalized entry 2:

```
2.  sem=the  rank=0.010000  num_args=1  pos=det
       phon=dh ah #    projection=det''
       usage=2  num_args=1  ifeats=[PER=3 DEF=Y ]
       arg 0  status=Mandatory  feats=[SUB=NOUN ]
```

The important detail to note is that the lexical acquisition procedure has applied a version space technique to the features of *the* resulting in a lexical entry which contains only the inherent features of the determiner, i.e. it subcategorizes for a noun and has the internal features PER=3 DEF=Y (3rd person and definite).

Now consider the effect of sentences 3-4, both of which involve the verb *bite* (only the entries for the verb are given here) and the resulting lexical entry, 5, which the lexical acquisition procedure hypothesizes:

```
3. I bite.

[WORD:[b aa1 iy t] POS:verb SEM:bite
ARGS:[[TH=AGT PER=1 NUM=SG DEF=Y ROLE=SUB] ] IFEATS:[TNS=PRS MOOD=IND] ]

4. A dog bites two men.

[WORD:[b aa1 iy t s] POS:verb SEM:bite
ARGS:[[TH=AGT PER=3 NUM=SG DEF=N ROLE=SUB] [TH=PAT PER=3 NUM=SG DEF=N ROLE=OBJ]]
IFEATS:[TNS=PRS MOOD=IND] ]

5.  sem=bite  rank=0.010000  num_args=2  pos=verb
       phon=b aa iy t #    projection=verb''
       usage=2  num_args=2  ifeats=[TNS=PRS MOOD=IND ]
       arg 0  status=Mandatory  feats=[TH=AGT ROLE=SUB ]
       arg 1  status=Optional   feats=[TH=PAT PER=3 NUM=SG ROLE=OBJ ]

    affix: 0  context=b aa iy t #
       changes=1  usage=1  rank=0.005000  pos=verb
       feats=[[CH=AFEAT0 PER=3 NUM=SG DEF=N ] [CH=AFEAT1 DEF=N ] ]
       old phonemes=#
       new phonemes=s #
```

[2] This phonetic string is converted into a distinctive feature [CH68] representation internally, in order to be able to precisely discern which phonetic features condition particular affixes. An affix might only apply to environments which are unvoiced, nasalized, coronal, etc.

The word *bite* now consists of a root, with the phonetic structure *b aa iy t* (= bite), and an affix, which adds the *-s* affix to the end of the word (indicated by #). The affix also adds the features PER=3 NUM=SG DEF=N (3rd person, singular, definite) to the verb's first argument feature list. After seeing more examples of *bite*, as well as other verbs which take the *-s* affix, the set of features which the affix contributes will be honed down to the proper set: PER=3 NUM=SG.[3]

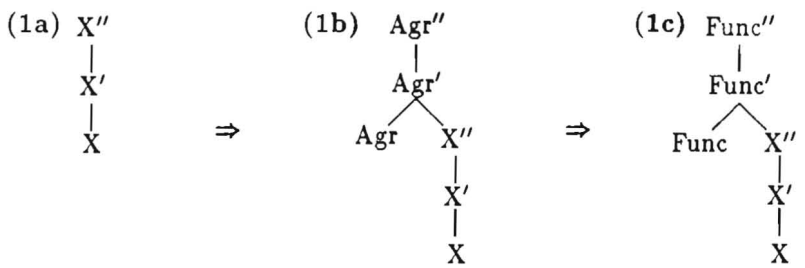
Once an affix has been created, it is free to combine with *any* word of the appropriate category, if that word provides the correct phonological environment (for instance, the *-s* affix must agree in voicing with the phonological material to which it immediately attaches). Affixes, as free agents, can then compete for use among the words of the language, and will be reinforced by any input word in which they appear. In this way, the productive affixes of the language will be identified.

# 3 Ramifications for Syntax

One way of viewing the lexical acquisition process just described is that it is a way of distilling information contained in the lexicon into meaningful classes: the roots and productive affixes of each of the lexical categories. This has two important consequences: 1) the affixes on a category signal that category's syntactic agreement relations; 2) if we adopt the additional assumption that a category's agreement information has an independent instantiation in *syntax*, as has been argued for theoretic and cross-linguistic reasons by Everett [Eve89], Abney [Abn87] and others, and for developmental reasons by Kazman [Kaz90] then this position provides just the environment necessary to analyze the functional categories of a language. In this way, the idiosyncratic structure of a language (as opposed to what is universal) may be determined through an examination of the lexicon.

Functional categories—things like determiners, complementizers and auxiliaries—tend to be phonologically and morphologically dependent, stressless and lack independent reference—they merely modify the meaning of their hosts, just like affixes. Furthermore, the information expressed by a function word in one language is often expressed by an affix in another. This lack of a consistent syntactic expression across languages argues against the treatment of functional categories as distinct categories universally—they are merely annotations to the meanings of the thematic categories. Given this array of facts, it seems natural to propose that the function words and affixes related to a particular category occupy the same agreement node position in the syntax.

This model assumes that the child initially projects all lexical items identically according to X′ theory and has no representation for FCs, as in (1a). Information about each category is learned by the child through a lexical acquisition process, which links the rate of acquisition of lexical items to input frequency and phonological salience. By analyzing the agreement properties of each category, the child will gradually learn which categories exhibit regular agreement processes—predictable meaning changes paired with changes in the phonetic form of a category. For these categories, he will posit an agreement node (Agr), dominating the lexical category, as in (1b). Finally, if the child hears function words which are manifestations of a category's agreement features, then these words will be identified with that category's Agr position, as indicated in (1c) by the re-labelling of the Agr position as *Func*.




Although these constructions are proposed as models of the child's development, each of the stages (1a-c) is a valid stage of Universal Grammar (UG). This can be stated confidently because examples of each kind of construction exist in the languages of the world: there are lexical categories which have no agreement (Adjectives in English, or any category in Chinese or Japanese), and would be represented as (1a). There are categories which exhibit agreement but contain no function words (nouns in Polish, for example), and would be represented as (1b). Finally, there are categories which both exhibit agreement and contain function words (like Nouns and Verbs in English, or

[3]The lexical entries shown here would not, in fact, be built after so few exposures to input. There is a built in conservatism in the acquisition system. This serves two purposes: it correctly models the child's conservatism and it allows the system the time and large numbers of exposures needed to accurately learn the intrinsic features of words.

any Romance language), and would be represented as (1c).

The model of syntactic projection works as follows: the lexical acquisition process, given an input of sentences accompanied by a semantic representation, distills categorial information into the roots and productive affixes of each of the lexical categories. When a category exhibits regular agreement affixes in the lexicon, an Agr node is hypothesized for that category in the syntax. Finally, if that category is modified by independent function words, then the Agr node provides a place wherein those words may be analyzed. This model cannot be "tricked" into making overgeneralizations about the language, because its basis of knowledge is generalizations made from the entire lexicon, and not a particular input word or sentence. By allowing the input to dictate which categories will be simple projections of the head (as in (1a)), which ones will exhibit syntactic agreement (as in (1b)) and which ones will contain function words (as in (1c)), a model of the language can slowly be built by the acquisition process. This method provides a way to "tailor" a maximally general grammar (the Universal Grammar) so that it will be able to adequately represent a particular language.

Consequently, the analysis of the lexicon provides a means by which the child can analyze function words in his grammar: when the child sees that a category utilizes a set of productive affixes, he instantiates a agreement node for these affixes in the syntax. This corresponds to the observed facts, as noted by language acquisition researchers—that function words become productive at the same time as children begin to master inflection.

# 4 Conclusions

I have presented a model of lexical acquisition which classifies the major thematic categories of a language according to whether they undergo productive agreement processes. This, in turn, allows syntactic structures to be built to allow a parser to correctly parse the language, including providing an environment in which function words may be analyzed. This is a fundamental characterization of a language, since, while languages typically share the same thematic categories, they differ widely on their use of functional categories (function words and agreement).

Furthermore, this model predicts that the set of affixes on a category, manifested as the category's agreement node, will be acquired at the same time as function words in the child's grammar. That is, the structures—agreement nodes—needed to analyze function words will be developed at this stage. This characterization has been shown to have important predictive ramifications cross-linguistically for English, French, Polish, Hebrew and Dutch [Kaz90].

This lexical acquisition mechanism has been implemented as a C program, and shown to be capable of making the necessary lexical generalizations to support this process. It currently is able to acquire the lexicon and project the syntactic structures necessary for English. The program is currently being tested with the acquisition of Polish, and results of this endeavor will also be presented at the conference.

# References


[Abn87] Steven P. Abney. *The English Noun Phrase in its Sentential Aspect.* PhD thesis, Massachusetts Institute of Technology, 1987.

[CH68] Noam Chomsky and Morris Halle. *The Sound Pattern of English.* Harper and Row, New York, 1968.

[Cho81] Noam Chomsky. *Lectures on Government and Binding.* Foris, Dordrecht, Holland, 1981.

[Eve89] Daniel Everett. Clitic doubling, reflexives and word order alternations in Yagua. *Language*, 65(2):339–372, 1989.

[GN88] Eithne Guilfoyle and Maire Noonan. Functional categories and language acquisition. Presented to the Boston University Conference on Language Acquisition, 1988.

[Kaz88] Rick Kazman. Null subjects and the acquisition of Case and Infl. Presented to the 13th Boston University Conference on Language Development, 1988.

[Kaz90] Rick Kazman. The induction of the lexicon and the early stages of grammar. Unpublished ms., Carnegie Mellon University, 1990.

[Mit77] T. M. Mitchell. Version spaces: A candidate elimination approach to rule learning. *IJCAI*, 5:305–310, 1977.

[PS87] Carl Pollard and Ivan Sag. *An Information-based Syntax and Semantics.* CSLI, Menlo Park, CA, 1987.

# Machine Learning and Language Acquisition

**Pat Langley**
AI Research Branch (MS 244-17)
NASA Ames Research Center
Moffett Field, CA 94035 USA
(LANGLEY@PTOLEMY.ARC.NASA.GOV)

## Abstract

In the early days of machine learning, language acquisition was a major focus of research. Recent work has focused on other topics, but many current issues are closely related to previous work on language learning. For instance, one current issue involves the extension of domain theories, which often take the form of Horn clause grammars, and one can view many early grammar-learning systems as addressing precisely this task. Another open problem concerns the generation of higher-order terms to improve induction, and a number of methods for grammar induction tackle this directly, rewriting sentences using such terms during parsing and constructing new ones when the existing grammar is inadequate. In addition, research on language acquisition can benefit from recent advances in other areas of machine learning. For example, methods for learning in problem solving, which provide algorithms for improving the efficiency of parsing, and techniques for concept formation offer incremental approaches to learning concepts with which one can later associate words. In general, researchers interested in both linguistic and nonlinguistic aspects of learning would benefit from closer inspection of each others' work.

## Personal history

I began working on grammar learning in the late 1970's, developing a model of first language acquisition that accounted for a number of phenomena observed in childrens' grammatical behavior. After some years of work in the area, I decided that all existing models relied too heavily on hand-crafted representations of the environment, and that before further progress could occur, we needed a model of concept formation in physical domains, preferably cast within a larger model of an intelligent agent. I have been actively working in this area since the middle 1980's, developing ICARUS, an integrated architecture that supports planning, perception, and action. Eventually, I hope to return to research on grammar acquisition, using ICARUS as the foundation. Along the way, I have also done some work on representation change, which bears a close relation to grammar induction, and I have written overview papers on the topic of language learning.

## Relevant publications

Langley, P. (1980). A production system model of first language acquisition. *Proceedings of the Eighth International Conference on Computational Linguistics* (pp. 183–189). Tokyo, Japan.

Langley, P. (1982). A model of early syntactic development. *Proceedings of the 20th Annual Conference of the Society for Computational Linguistics* (pp. 145–151). Toronto, Ontario.

Langley, P. (1982). Language acquisition through error recovery. *Cognition and Brain Theory*, *5*, 211–255.

Langley, P. (1987). A general theory of discrimination learning. In D. Klahr, P. Langley, & R. Neches (Eds.), *Production system models of learning and development.* Cambridge, MA: MIT Press.

Langley, P. (1987). Machine learning and grammar induction. *Machine Learning*, *2*, 5–8.

Langley, P., & Carbonell, J. G. (1987). Language acquisition and machine learning. In B. MacWhinney (Ed.), *Mechanisms of language acquisition.* Hillsdale, NJ: Lawrence Erlbaum.

# Machine Learning and Language Acquisition

**Pat Langley**
AI Research Branch (MS 244-17)
NASA Ames Research Center
Moffett Field, CA 94035 USA

## Introduction

Although research on language acquisition has a long history within the field of machine learning, in recent years attention has focused mainly on other topics. However, many recently 'discovered' issues relate directly to problems in the acquisition of linguistic knowledge, and in some cases, tentative solutions already exist in the literature on language learning. In other cases, research in other areas of machine has important implications for work on language acquisition, although this may not be apparent at first glance.

In this paper, I explore some relations between these superficially different aspects of the learning process. The organization follows four topics that are currently popular within the machine learning community – extending domain theories, representation change, learning in problem solving, and concept formation. In the first two cases, I consider early work on language acquisition that is relevant to current issues. In the latter two cases, I consider some learning methods developed for other purposes that may aid in developing methods for language learning. In all cases, I emphasize the underlying unity of issues that arise in machine learning, whatever the domain of application.

## Extending domains theories

One of the most active paradigms within machine learning focuses on *explanation-based* approaches (Mitchell, Keller, & Kedar-Cabelli, 1986; DeJong & Mooney, 1986). In this framework, the learner begins with some background knowledge, or *domain theory*, often stated as inference rules or Horn clauses. Upon encountering a positive training instance $I$ for a concept $C$, one uses the domain theory to explain why $I$ is an example of $C$. Typically, this explanation consists of a logical proof, although less formal approaches are possible. One then uses the explanation to identify relevant descriptors in the instance, along with variable bindings shared among descriptors. After this, one constructs a new inference rule that lets one infer the concept directly from these descriptors, without the intermediate steps in the explanation.

Early work on explanation-based learning assumed that the domain theory was complete and correct. These simplifications were natural to make, in that these methods compiled existing knowledge into another form, making the accuracy of the learned rules a direct function of the accuracy of the background knowledge. However, researchers realized from the outset that they would eventually have to relax these assumptions, and methods for extending incomplete domain theories and revising incorrect ones are active topics of research (e.g., Hall, 1988; Ourston & Mooney, 1990; Pazzani, 1989). In most cases, this work uses the domain theory to construct a partial explanation of training instances, then uses descriptors in the instances as material for altering the domain theory.

Few researchers in this tradition have noted the close connection to earlier work on grammar acquisition. For instance, Anderson (1977) and Wolff (1982) describe methods for using existing grammatical rules to construct partial parses, and then extending the grammar based on the words occurring in the training sentence. Langley (1982) and Reeker (1976) take a very similar approach to extending grammars for generation, using existing rules to produce partial sentences, comparing these to ones uttered by a teacher, and using differences to suggest extensions. All of these systems represent grammatical knowledge in ways that can be mapped directly onto the inference rules used in most work on explanation-based learning. The partial parses these programs construct can be viewed as partial explanations, and the revision of grammars can be viewed as the revision of incomplete or incorrect domain theories.

The relation between these two research efforts has been missed because of differences in terminology, area of application, and – most important – rhetorical stances. Most work on grammar induction has emphasized the inductive nature of this task, and has paid little attention to intermediate states in which the learner has only a partial grammar. In contrast, research on explanation-based learning has emphasized the importance of 'justified' learning, and has argued against the use of inductive methods. However, despite this rhetoric against empirical approaches, the alter-

ation of domain theories – like grammar acquisition – is an inherently inductive task, in that it requires one to move beyond information in the training instances. Still, there is room for building on existing knowledge, whether one refers to this knowledge as an incomplete domain theory or as a partial grammar.

## Representation change

Another important issue within machine learning concerns constructive induction and representation change. The first of these deals with the need to rewrite instances using higher-level terms, so that induction can occur in this more appropriate language. For instance, given the set of positive instances {1, 15, 3, 29, 7} and negative instances {4, 12, 6, 28}, a trivial rule suggests itself to those familiar with the notion of even and odd numbers. Without this, no regularity is apparent, and the most one can do is form a disjunctive concept based on the observed instances. Recent work by Drastal, Czako, and Raatz (1989), Elio and Watanabe (in press), and others have shown that constructive induction can increase both the rate of learning and asymptotic accuracy.

Research on representation change focuses on the generation of such higher-level terms from experience. The idea here is to detect regularities in the training instances, introduce new terms that summarize these regularities, and use these terms to simplify the induction later in the learning process. Matheus (1989) gives an insightful review of work in this area by Muggleton (1987), Schlimmer (1987), and many others. This research paradigm has shown that the introduction of new terms can lead to improved learning.

Despite progress in this area, few researchers have recognized that issues arising in representation change are closely linked to ones that arise in grammar induction. In many cases, one can view the process of rewriting an instance in higher-level terms as a form of parsing, and one can cast the act of creating new terms as the induction of new word classes and phrases. In fact, some existing grammar induction systems deal directly with these issues, parsing new sentences as they are observed and constructing new terms when the existing grammar is inadequate.

For example, consider Wolff's (1982) SNPR algorithm, which induces phrase-structure grammars from sequences of letters given as input. The system carries out a hill-climbing search through the space of such grammars, using two basic operators. The first notes frequently occurring sequences of symbols and defines new 'chunks', which correspond to words and phrases. The second learning operator notes when sets of symbols tend to occur in the same context (i.e., next to a common symbol); this defines new disjunctive classes, which correspond to parts of speech and alternative forms of phrases.

SNPR is semi-incremental, in that it processes only part of its input at a given time, using the terms it introduces during earlier learning in processing its later experience. Specifically, the system constructs a partial grammar to summarize the letter sequences it has observed, and then it uses this grammar to rewrite new strings at a higher level of description (i.e., using nonterminal symbols in the grammar). This is isomorphic to the process used in constructive induction to redescribe training instances, and the process of defining new chunks and classes is clearly a form of representation change.[1]

Other grammar acquisition systems, such as Siklóssy's (1972) ZBIE and Anderson's (1977) LAS, also introduce conjunctive terms (for phrases) and disjunctive terms (for word and phrasal classes). In these cases, the focus is on inducing mappings between sentences and their meanings. This provides additional constraints on the learning process, so that ZBIE and LAS can rely less on the type of distributional information used by SNPR. However, the types of learned knowledge structures play a similar role, and both systems can be viewed as carrying out constructive induction. Machine learning researchers interested in this topic would do well to study this early work on grammar learning.

## Learning in problem solving

Most work on problem solving within AI operates within the paradigm of search through some problem space. At each step in this search, one must select an operator to apply to some problem state, which generates a new state from which the search continues. The combinatorial nature of most problem spaces can be constrained by heuristics, which suggest operators or states to select. Thus, one obvious role for learning within this framework is to acquire such search control knowledge. Laird, Rosenbloom, and Newell (1986), Langley (1985), Minton (1990), and many others have taken this approach to learning in problem-solving domains. An alternative approach is to acquire macro-operators, which let one take many steps through the problem space in a single leap. Iba (1989), Shavlik (1990), and others have explored this approach, most involving some form of explanation-based learning.

Many treatments of parsing note the importance of search in understanding sentences, an issue that cuts across different representations of linguistic knowledge. In augmented transition networks, one must decide which arc to consider at each node. In phrase-structure grammars, one must decide which rewrite rule to use in expanding a symbol. Machine learning methods for

[1] One can also view SNPR as extending an incomplete domain theory. At each stage in learning, the system uses its existing grammar to construct partial parses, then extends the grammar based on observed letter sequences.

reducing search have important implications for the parsing task, although few researchers from either the language or the learning community have noted this potential. This is probably because traditional work on language learning has focused on the acquisition of *accurate* grammars, rather than *efficient* ones.

However, there are some exceptions to this rule. For instance, Carlson, Weinberg, and Fisher (1990) have recently used an inductive learning technique to improve search control (and thus parsing efficiency) in network grammars. Also, Rayner (1988) has applied explanation-based methods to compile macro-operators that improve the efficiency of parsing based on rewrite rules. The learning task of improving parsing efficiency has much to recommend it for both language and learning researchers. The problem is well defined, there exist clear performance criteria, and there now exist many large grammars (i.e., domain theories) to support work in the area.

Still, improved parsing efficiency is not the only application of search-related methods to language learning. Consider Berwick's (1985) approach, which represents grammatical knowledge as a set of production rules. In his framework, actions involve parsing operators such as creating a node in a parse tree, putting a node in an input buffer, attaching a node to a partial parse tree, and switching items in the buffer. Thus, parsing can be viewed as a state-space search, in which the goal is to produce a complete parse tree and an empty input buffer. His acquisition system attempts to parse new sentences using these operators, invoking background knowledge to eliminate illegal steps, and using steps along the solution path as positive training instances. The system then carries out induction over these training instances to determine the legal conditions for applying each parsing operator. This approach is very similar to work on heuristics learning for state-space problem solving (e.g., Langley, 1985).

## Concept formation

Most research on language acquisition has dealt with grammar learning, but there has been some work on the acquisition of word meanings, which has taken two basic approaches. The first assumes that symbols for the relevant concepts already exist in long-term memory, and all that remains is to link words to the appropriate concepts (e.g., Siklóssy, 1972). The other scheme defines word meanings in terms of more primitive conceptual structures, but assumes that meaning acquisition is largely a supervised learning task (Salveter, 1979; Selfridge, 1981). However, there is evidence that, at least in many cases, children form useful concepts long before they attach words to those symbols. This suggests that the first approach provides a better view of human word learning, but existing models provide no explanation for the origin of concepts to which words are linked.

Recent work on *concept formation* – which Gennari, Fisher, and Langley (1989) define as the incremental acquisition of concepts from unlabeled instances – offers a path out of this dilemma. Techniques for concept formation interleave the process of classifying an experience and the process of altering memory to incorporate that experience. Over time, such methods build up a complex memory of concepts at different levels of abstraction, which can be used for recognition and prediction. Fisher and Langley (1990) argue that these methods provide useful models of human concept representation, use, and acquisition. Initial studies in this area focused on simple attribute-value domains, but more recent work has dealt with concepts that involve structure and change over time.

To date, research on concept formation has not addressed the problem of word meanings, but it provides a promising framework for future work in this area. Assumptions about the representation and organization of concepts from this paradigm provide constraints on approaches to meaning acquisition, and the latter provides a task that could challenge existing concept formation techniques. Extensions to existing mechanisms may prove sufficient to associate words with acquired concepts, giving a unified model of concept formation and the acquisition of word meanings.

## Conclusions

In summary, previous work on language acquisition has addressed a number of issues that are currently receiving attention within the broader machine learning community, and researchers in the latter tradition have much to learn from the former. Similarly, recent advances in nonlinguistic areas of machine learning have important implications for the study of language acquisition, and researchers interested in this topic would do well to examine work outside their own area. Many of these methods learn in an incremental manner, a prerequisite for modeling human learning and, indeed, for supporting any intelligent agent that must interact with an external environment over long periods of time.

In fact, the growing interest in constructing integrated architectures for intelligent agents may directly support research on language acquisition (Laird et al, 1986; Langley & Carbonell, 1987; Langley, Thompson, Iba, Gennari, & Allen, in press). A number of proposed architectures include learning mechanisms as central components, and the increasing concern with perception may overcome the hand-crafted representations of meaning assumed by many early models of linguistic learning, which bore a remarkable resemblance to parse trees. An integrated approach to cognition, perception, and action – the goal of research on architectures for intelligent agents – may provide the foundation required for a complete model of language acquisition.

## References


Anderson, J. R. (1977). Induction of augmented transition networks. *Cognitive Science*, *1*, 125–157.

Berwick, R. C. (1985). *The acquisition of syntactic knowledge*. Cambridge, MA: MIT Press.

Carlson, B., Weinberg, J., & Fisher, D. (1990). Search control, utility, and concept induction. *Proceedings of the Seventh International Conference on Machine Learning* (pp. 85–92). Austin, TX: Morgan Kaufmann.

DeJong, G., & Mooney, R. J. (1986). Explanation-based learning: An alternative view. *Machine Learning*, *1*, 145–176.

Drastal, G., Czako, G., & Raatz, S. (1989). Induction in an abstraction space: A form of constructive induction. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 708–712). Detroit, MI: Morgan Kaufmann.

Elio, R., & Watanabe, L. (in press). An incremental deductive strategy for controlling constructive induction in learning from examples. *Machine Learning*.

Fisher, D. H., & Langley, P. (1990). The structure and formation of natural categories. In G. H. Bower (Ed.), *The psychology of learning and motivation: Advances in Research and Theory* (Vol. 26). Cambridge, MA: Academic Press.

Gennari, J. H., Langley, P., & Fisher, D. H. (1989). Models of incremental concept formation. *Artificial Intelligence*, *40*, 11–61.

Hall, R. J. (1988). Learning by failing to explain: Using partial explanations to learn in incomplete or intractable domains. *Machine Learning*, *3*, 45–77.

Iba, G. A. (1989). A heuristic approach to the discovery of macro-operators. *Machine Learning*, *3*, 285–317.

Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). Chunking in SOAR: The anatomy of a general learning mechanism. *Machine Learning*, *1*, 11–46.

Langley, P. (1982). Language acquisition through error recovery. *Cognition and Brain Theory*, *5*, 211–255.

Langley, P. (1985). Learning to search: From weak methods to domain-specific heuristics. *Cognitive Science*, *9*, 217–260.

Langley, P., & Carbonell, J. G. (1987). Language acquisition and machine learning. In B. MacWhinney (Ed.), *Mechanisms of language acquisition*. Hillsdale, NJ: Lawrence Erlbaum.

Langley, P., Thompson, K., Iba, W., Gennari, J. H., & Allen, J. A. (in press). An integrated cognitive architecture for autonomous agents. In W. Van De Velde (Ed.), *Representation and learning in autonomous agents*. Amsterdam: North Holland.

Matheus, C. J. (1989). *Feature construction: An analytic framework and an application to decision trees*. Doctoral dissertation, Department of Computer Science, University of Illinois, Urbana-Champaign.

Minton, S. N. (1990). Quantitative results concerning the utility of explanation-based learning. *Artificial Intelligence*, *42*, 363–391.

Mitchell, T. M., Keller, R. M., & Kedar-Cabelli, S. T. (1986). Explanation-based generalization: A unifying view. *Machine Learning*, *1*, 47–80.

Muggleton, S. (1987). DUCE: An oracle-based approach to constructive induction. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence* (pp. 287–292). Milan, Italy: Morgan Kaufmann.

Ourston, D., & Mooney, R. J. (1990). Changing the rules: A comprehensive approach to theory refinement. *Proceedings of the Eighth National Conference of the American Association for Artificial Intelligence* (pp. 815–820). Boston, MA: AAAI Press.

Pazzani, M. J. (1989). Detecting and correcting errors of omission after explanation-based learning. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 713–718). Detroit, MI: Morgan Kaufmann.

Rayner, M. (1988). Applying explanation-based generalization to natural-language processing. *Proceedings of the Conference on Fifth Generation Computer Systems*. Tokyo.

Reeker, L. H. (1976). The computational study of language acquisition. In M. Yovits & M. Rubinoff (Eds.), *Advances in computers* (Vol. 15). New York: Academic Press.

Salveter, S. (1979). Inferring conceptual graphs. *Cognitive Science*, *3*, 141–166.

Schlimmer, J. C. (1987). Incremental adjustment of representations for learning. *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 79–90). Irvine, CA: Morgan Kaufmann.

Selfridge, M. (1981). A computer model of child language acquisition. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence* (pp. 92–96). Vancouver, BC: Morgan Kaufmann.

Shavlik, J. W. (1990). Acquiring recursive and iterative concepts with explanation-based learning. *Machine Learning*, *5*, 39–70.

Siklóssy, L. (1972). Natural language learning by computer. In H. A. Simon & L. Siklóssy (Eds.), *Representation and meaning: Experiments with information processing systems*. Englewood Cliffs, NJ: Prentice Hall.

Wolff, J. G. (1982). Language acquisition, data compression, and generalization. *Language and Communication*, *2*, 57–89.

# A Model of the Development of Phrase-Structure

David LeBlanc
Department of Computer Science
University of British Columbia
Vancouver, B.C.
V6T 1Z2
leblanc@cs.ubc.ca

Position
PhD student in computer science

Research Interest
First Language Acquisition
Parsing
Psychoinguistics

Bibliography
"The Generation of Phrase-Structure Representations From Principles"
TR-90-38 Dept. of CS, UBC 1991

Parsing With Principles
Submitted to:
ACL 91

Henry Davis
Department of Linguistics
University of British Columbia
Vancouver, B.C.
V6T 1Z1
c/o leblanc@cs.ubc.ca

Position
visiting Assistant Professor

Research Interest
Syntax
Language Acquisition
Psycholinguistics

Bibliography
"The Acquisition of the English Auxiliary System and its Relation to Linguistic Theory"
PhD Dissertation, UBC Dept. of Linguistics 1987

"The Acquisition of the English Auxiliary System and its Relation to Linguistic Theory"
To be published by Cambridge University Press

**Abstract**

The design of a computational first language acquisition model requires three steps: 1) identify a satisfactory theory of acquisition, 2) show the validity of the theorized mature grammar (of the acquisition theory), and 3) construct a model of the theorized acquisition device. In this paper we will present a theory of acquisition which we feel is suitable for future computational development, thus satisfying step 1). Furthermore, we will present a computational model of the theorized mature grammar, satisfying step 2). And, although we introduce no explicit methodology for the satisfaction of step 3), we present introductory ideas for the implementation of an acquisition device which we hope will initiate discussion.

# 1 Introduction

To design a computational first language acquisition model, one must proceed through three steps:

1) determine an appropriate theory of acquisition, either by identifying an existing theory or postulating one,
2) design and implement a computational model of the theorized mature grammar to show that the end result of the acquisition process can be achieved, and
3) design and implement a computational model of the acquisition system which will achieve the theorized mature grammar.

In this paper, we will describe a project in which we have completed the first two steps, and are about to actively pursue research towards meeting the third.

# 2 The Theorized Mature Grammar

The acquisition theory chosen to be the framework of this project is Davis' version of Government and Binding Theory. Presented as a dissertation in 1987 (at the University of British Columbia, Dept. of Linguistics) this theory has been designed explicitly as a model of acquisition. In the introduction to the dissertation, Davis argues from Chomsky's definition of epistemological priority that a system which maps prelinguistic primitives into a linguistic theory is preferable to one defined solely as a linguistic or prelinguistic model. With this goal in mind, he presents a theory in which the traditional generation of phrase-structure representations from rules (whether the explicit phrase-structure rules of context-free rule-based systems and unification-type grammars or the highly generalized Xbar-theory of GB) is replaced by four simple principles of node domination which determine the categorial features of the dominating node of any two sister nodes in a representation tree using Case, Theta, and categorial information. It is argued that this mapping of prelinguistic information onto a linguistic representation (tree structures) meets the definition of epistemological priority and is therefore preferable to explicit phrase-structure rule-based systems.

In order to derive these percolation principles, Davis has had to rework many of the traditional components of GB Theory. He starts by dividing the supposedly unifying concept of government into two distinct forms, internal and external. Internal government concerns the relationship between a lexical governor and elements within its maximal projection. External government concerns the relationship between a governor and the elements within a maximal projection it governs. Internal government is further divided into its core case in which a lexical head governs its complements (in a 'Canonical Government Configuration (CGC)' - minimal government) and maximal government which corresponds to the m-command of Chomsky (1986). External government is a more murky concept which defines government down a tree structure as dictated by barriers to government and is beyond the scope of this abstract. It suffices to distinguish our two types of government:

Minimal Government

> A minimally governs B iff A minimally c-commands B and there is no C such that A governs C and C governs B.

Maximal Government

> A maximally governs B iff A maximally c-commands B and there is no C such that A governs C and C governs B.

Davis uses the core case of internal government to define internal Theta-assignment. Just as government is divided into internal and external cases, Theta-theory is defined in terms of internal Theta-assignment (which only occurs within a CGC) and external Theta-assignment. External Theta-assignment relies upon predication which stipulates how predicates are linked to their external arguments (i.e., subjects). This relationship is defined by the relationship between the verb and its AGR-bearing INFL, and Case assignment by INFL to the subject. Davis also posits the elimination of the controversial PRO by the relaxation of the Theta Criterion to allow arguments to bear more than one Theta-role.

Case theory has also been divided into internal and external assignment. Internal Case-assignment normally takes place in a CGC, although in certain circumstances it is able to penetrate a derived XP to exceptionally Case-mark its specifier. External Case (in English, leftward Case) is the assignment of Case to the subject. It differs from internal Case in level of application, obligatoriness, and (in English) direction and adjacency conditions.

Given these definitions of government, Theta-theory and Case-theory, we can now present the first two of the four Percolation Principles:

Percolation Principle I

> Where X Theta-governs Y, the categorial features of Z (the dominating node) will be those of X.

Percolation Principle II

> Where X assigns Case to Y, the categorial features of Z will be those of X.

Percolation Principle III deals with the difference between the adjunction set and the subcategorization set of a phrase. We note that adjuncts, elements which are not tied to others by Theta- or Case- relations, typically have no effect on categorial structure. In other

words, the categorial features of a node dominating a member of the adjunction set will bear the features of the other node to which it is joined. In order to define a principle based upon this observation, we need to formally differentiate between adjuncts and members of the subcategorization set. This can be done by modifying the Revised Extended Projection Principle of Chomsky (1982) to what Davis calls the GREPP.

Generalized Revised Extended Projection Principle

> Subcategorization requirements must be satisfied by all phrase- structure configurations, where "subcategorization requirements" refer both to subcategorized and subcategorizing elements.

This definition has the effect of extending the concept of subcategorization to include both the subcategorizing and subcategorized elements. We may now define Percolation Principle III as:

Percolation Principle III

> Where X is a member of the adjunct set and Y a member of the subcategorization set of a phrase Z, the categorial features of Z will be those of Y.

Finally, we need a principle which determines the categorial features of a dominating node if none of the above conditions is present. To this end, Davis presents a percolation hierarchy based upon three types of categories he introduces to capture categorial generalizations: Theta-heads (N,V,A), G-heads (INFL, Det) and C-heads (complementizers and prepositions). These types of categories enter into categorial associations with one another (based on the notion of functional discharge) in all of the Percolation Principles, but most importantly in IV.

Percolation Principle IV

> Where X and Y are in a CGC, no Case or Theta relation holds between them, and both are part of the subcategorization set of Z, the following hierarchy determines which features will percolate:
> a. C-features of X and Y will percolate to Z
> b. G-features of X and Y will percolate to Z
> c. Theta-features of X and Y will percolate to Z

# 3 Psychologically Plausible Language Acquisition

In principle, in order to achieve psychological plausibility (see Pinker 1979) a model of human first-language learning must successfully account for the acquisition of any possible human grammar, within the time-span in which normal first-language learning takes place, given the input available, and what is known about the cognitive abilities and limitations of young children; moreover, since learning is non-instantaneous, such a model must mimic the course of real-time acquisition by predicting where learners make mistakes and the order in which they acquire syntactic rules and representations.

Given out present knowledge, this is a tall order; nevertheless, we feel that recent progress in theoretical, computational and psychological approaches to language has put us in a position to make tentative proposals concerning the structure of such a model. In this section we will begin by presenting a 'logical' version of (idealized) language acquisition; we will then examine the contribution of data from real-time language acquisition to the issues under discussion; and we will end by proposing some significant modifications to the model, designed to increase its 'psychological plausibility'.

A viable model of language acquisition must contain the following component:

(i) A theory of the target grammar.
(ii) A specification of the input (Primary Linguistic Data, henceforth PLD).
(iii) A learning mechanism.

We will take as our target a government-binding type grammar as described previously.

As for input, we will adopt the following assumptions:

(i) Young children receive and employ no negative evidence (i.e., their utterances are not generally corrected for grammaticality, nor do they attend to such corrections if offered).
(ii) Input is 'noisy' - it contains slips of the tongue and incomplete and fragmentary utterances.
(iii) The child is limited in its linguistic 'intake' by independent cognitive constraints connected with short-term memory, sequencing, and lexical retrieval.

While not entirely free from controversy, these three assumptions are generally well-supported empirically in the child-language literature. Together they constitute the basis of the enormously influential "poverty of stimulus" argument which has informed much work in the so-called 'logical' theory of language acquisition. This theory is based upon the premise that powerful innate constraints must be operative during language learning in order to ensure that the child identifies a target grammar on the basis of inadequate PLD. In its most extreme version, embodied in the 'principles-and-parameters' model of Chomsky and his followers (1981 and elsewhere), grammar-learning is reduced to a choice between a few abstract parameters 'triggered' by certain key types of data readily available in the input.

Let us then turn to the learning mechanism. Surprisingly little attention has been paid to this part of the acquisition theory until recently. It has been generally assumed that the child, innately equipped with a rich deductive system (Universal Grammar, henceforth UG) "learns" by hypothesis testing. If the child encounters sentences in the target language which are not generated by his or her grammar, UG will alter a parameter setting; the resulting grammar will then once again be checked against the input, and altered further until no counter-evidence is encountered. At this point the child 's Language Acquisition Device will have converged on the adult grammar. It should be pointed out that under this conception, the relationship between UG and the PLD is indirect; parameter settings are ordered by UG according to a fixed and innately predetermined hierarchy, constrained by considerations of cross-linguistic markedness and by the Subset Principle of Berwick (1985) and much subsequent work. The latter ensures that a child will never guess at an overgeneral grammar, and then be forced to backtrack; given the proscription against negative evidence mentioned above, retreat is theoretically impossible for the first-language learner.

The logical theory of language acquisition, as briefly described above, is the first viable 'non-instantaneous' theory of grammar learning. It provides a solution to the problem posed by the poverty of stimulus argument, by severely constraining the learner's hypothesis space; in fact, 'learning' is reduced to choosing between a few limited parametric alternatives. However, the logical theory is less successful at accounting for the observed course of real-time language acquisition. This is not surprising, since it was not designed to do so; nevertheless, it provides a useful null hypothesis against which more psychologically plausible models can be judged.

A parametric model based on logical acquisition makes the following predictions for real-time acquisition:

(i) Less marked grammars will be initially hypothesized by the child, regardless of the relative markedness of the target grammar. Early grammars will correspond to cross-linguistically unmarked systems.
(ii) Due to the operation of the Subset Principle, less inclusive grammars will be hypothesized before more inclusive ones; the child will not overgeneralize, since retreat is impossible.
(iii) Because setting a single parameter can have multiple effects on the resulting grammar, we should expect to find distinct 'stages' in language acquisition corresponding to different parameter-settings and characterized by sets of parametrically linked syntactic properties.
(iv) Since UG is a shared genetic component, and since the PLD available to children is relatively uniform, we should expect the course of acquisition to be similar across subjects learning the same language.

None of these predictions are borne out by evidence from language acquisition. In fact, the following generalizations seem to characterize the acquisition process:

(i) While there has been some work claiming that early stages in acquisition correspond to less-marked parameter settings (see in particular Hyams (1986) on the pro-drop parameter in Italian and early English) the expected strict correlation between syntactic markedness and relative ease of acquisition has failed to emerge. To give an example, preposition stranding, as in (1a), as opposed to "piedpiping", exemplified in (1b), is known to be marked cross-linguistically:

1a. What are you talking about?
 b. About what are you talking?

Yet English-speaking children acquire structures like (1a) literally years before those like (1b) (see French 1984). Of course, this is to be expected given the PLD available to English-learning children, who are far more likely in colloquial speech to encounter stranded structures than their stylistically-marked pied-piped equivalents. However, in an unmodified parameter-setting model, this is irrelevant; relative frequency of structures in the input should make no difference to the invariant and innately ordered sequence of hypotheses available to the learner.
(ii) The Subset Principle predicts no overgeneralization; yet a pervasive pattern of overgeneralization characterizes the acquisition of certain syntactic elements. It will be argued below that these form a natural class - that of 'closed-class' or 'functional' elements - and that such elements are associated with a particular type of input-sensitive learning.
(iii) Again contrary to the predictions of an unmodified parametric model, acquisition is typically uneven and variable both across and within categories. Note that a "lexicalized" parametric model such as that suggested by Wexler and Manzini (1987) does not solve this problem, since it neither accounts for why general patterns eventually emerge, nor for the fact that the same form may be produced in more than one way at the same time.
(iv) One of the most striking conclusions to emerge from the child language literature is the surprising extent of individual variation in linguistic development. If neither the input nor the Language Acquisition Device is variable, the parametric model has simply no way of accounting for such variation.

Thus a logically feasible model of language acquisition cannot translate straightforwardly into a psychologically plausible one. At the same time, it should be pointed out that the latter must retain the advantages of the former: there is no point in modelling real-time language acquisition if the model cannot attain the end-point of a stable human grammar, given the available input. And

of course, the original poverty of stimulus argument still holds.

What all this suggests is that two different types of learning mechanism must be available to the language learner. On the one hand, a powerful deductive system is needed to account for the successful acquisition of target grammars which are severely underdetermined by the PLD; on the other, an inductive, data-sensitive mechanism seems necessary to account for the complex patterns of over- and under- generalization actually observed during language development.

There is in fact some intriguing empirical evidence in favour of such a suggestion. Newport, Gleitman and Gleitman (1977) discovered (as part of their study on the relationship between properties of the PLD and language learning) a correlation between the acquisition of auxiliary elements in (canonical) medial position and the presence in the input of fronted auxiliaries (in questions). This correlation was subsequently confirmed by several other similar studies (see, for example Furrow, Benedict and Nelson 1979 and Newport, Gleitman and Gleitman 1984). It thus appears that the acquisition of auxiliary elements, which are generally unstressed and often contracted in medial position, is dependent on their appearance in the more salient fronted position. Yet when it comes to production, children initially use auxiliaries only in medial position, failing to invert them in both yesno and WH-questions. This leads to a curious and paradoxical situation: learners need input containing fronted auxiliaries, yet they initially seem unable to produce auxiliaries in fronted environments.

A similar situation seems to hold in the acquisition of German. German is underlyingly an SOV language, but in main clauses an obligatory rule fronts an inflected verb, leading to a surface word order where the verb either occupies second position (in declaratives) or first position (in yes-no questions). Under standard assumptions, this derived word order will overwhelmingly predominate in the input to young children, which is generally monoclausal. Yet the initial word order produced by German speaking children is apparently almost always verb-final; it is only when verbal inflections are acquired that word-order reflects the verb-second constraint (see Clahsen 1984, Mills 1984). Once again, we are forced to the odd conclusion that learners of German 'undo' inflected-verb movement to get at a basic word order, but then are unable to re-apply it productively until further developments have taken place in the grammar.

What are these further developments? In Davis (1987), it is argued that in both the English and German cases, correct production of the relevant structures is dependent on the acquisition of certain inflectional elements connected with syntactic agreement. Generalizing from these cases, Davis proposes a 'two-tiered' model of language acquisition. The first tier consists of a 'recognition' phase, in which the child makes use of the deductive power of UG to set basic parameters and establish fundamental structural properties of the language to be learnt. The second tier involves an 'instantiation' phase, in which the child must acquire specific functional elements before being able to use his or her syntactic knowledge in production.

These two tiers, moreover, involve quite different types of learning mechanism. Recognition-type learning is extremely general, involving abstract syntactic categories; and extremely successful, in that there is little or no evidence for difficulties in the acquisition of fundamental syntactic properties of a language, such as the identification of basic grammatical categories and relations. On the other hand, instantiation is frequently error-laden, and shows a cluster of characteristic properties, including

(a) A 'U-shaped' learning curve, involving early undergeneralization, subsequent over-generalization, and final retreat.
(b) 'Lexical learning effects'.
(c) Type and token variation in production.

This pattern is also characteristic of the acquisition of inflectional morphology (see McLelland and Rumelhart 1987). Davis (1987) claims, following Emonds (1985), that there is a syntactically and psycholinguistically significant distinction between open-class elements, which are learnt in a maximally general fashion, and closed-class elements, including functional elements such as auxiliaries, determiners and complementizers as well as inflectional morphemes, which are learnt in a highly specific, input-sensitive and probabilistic manner, accounting for properties (a-c) above. The exact nature of the closed-class learning mechanism is open to some debate; it is possible that it might involve either connectionist-type architecture, or a constraint-satisfaction type model, as suggested by Pinker (1987). It should be noted, however, that the role of such a mechanism is strictly limited in this model to 'low-level' learning; there is no suggestion that it could supplant the highly abstract and theory-laden mechanisms needed for syntactic recognition.

Thus the two-tiered approach allows us to develop a learning model which accounts in a principled manner for the child's ability both to generalize and to particularize during the course of language acquisition. It also meshes well with what is known about real-time language acquisition. It thus seems to be a promising approach to the construction of a model of acquisition which takes psychological plausibility seriously.

# 4 A Computational Model of the Mature Grammar

Having previously examined the model of mature grammar described by this theory, we can now describe how this was implemented in a computational parsing mechanism. Implementing the theory presented three major issues to be overcome:

1. the intrinsic right-left nature of the Percolation Principles,
2. multiple possible subcategorizations (and related Case and Theta assignments), and
3. movement.

The Percolation Principles, as described, are intrinsically right-left in nature as they presuppose knowledge of the categorial features of sister nodes. As we are positing tree structures as our representations, the rightward branch of any dominating node (other than the rightmost) will be the dominating node of the rest of the sentence. Therefore, we must know the structure of the representation of all rightward nodes before forming any dominating node. This is not a psychologically valid approach as it is almost certain that people process sentences left-to-right. We resolve this problem by observing that PPI does not necessarily require knowledge of the right- adjacent node. PPI stipulates that the categorial features of a a node which assigns a Theta-role to a right-sister node will dominate regardless of the features of the rightward node. As we have seen, internal Theta-assignment is always in a CGC, thus we can form a dominating node whenever we encounter an internal Theta-assigner. Once one dominating node is formed, a representation of all nodes thus far encountered can be constructed. This allows correct partial representations to be formed while processing left- to-right.

The fact that many predicates have more than one possible subcategorization type presents a problem in our left-right parsing paradigm. As we cannot know the dominating node of the actual subcategorized phrase until we have actually parsed it, we cannot choose the correct subcategorization type in advance. As the subcategorization itself can influence the phrasal type of the argument (eg., a CP headed by an empty C), we must actually try the different possible subcategorizations until a 'match' is discovered. We have found that ordering the possible subcategorizations speeds processing; the parser tries two element subcategorizations first, then PP, CP, IP and NP. This is however strictly an issue of implementation.

Movement within the sentence is handled using a filler-driven paradigm. When an element is encountered which does not receive the required Case and Theta assignments, movement is flagged and subsequent processing tests for possible gaps (corresponding to the moved element's original position in the sentence). Gaps are identified as positions which are assigned Case and/or Theta roles, but have no receiver present. Procedures have been implemented to handle leftward movement of arguments and non-arguments, but rightward movement remains problematic.

# 5 References


[**Berwick 85**] Robert C. Berwick, *The Acquisition of Syntactic Knowledge* , The MIT Press.

[**Chomsky 81**] Noam Chomsky, *Lectures on Government and Binding* , Foris Publications, Dordecht.

[**Chomsky 82**] Noam Chomsky, *Some Concepts and Consequences of the Theory of Government and Binding* , MIT Press, Cambridge, MA.

[**Chomsky 86**] Noam Chomsky, *Barriers* , The MIT Press, Cambridge, MA.

[**Clahsen 84**] H. Clahsen, *Spracherwerb in der Kindheit. Eine Untersuchung zur Entwicklung der Syntax bei Kleinkindern*, Narr, Tubingen.

[**Emonds 85**] J. Emonds, *A Unified Theory of Syntactic Categories* , Foris Publications, Dordrecht.

[**French 84**] M. French, "Markedness and the Acquisition of of Piedpiping and Preposition Stranding", in *McGill University Working Papers in Linguistics* 2: 131-145.

[**Furrow** *et al.* **79**] Furrow, Nelson and Benedict, "Mothers Speech to Children and Syntactic Development - Some Simple Relationships", *Journal of Child Language* 6: 423-442.

[**Hyams 86**] Nina Hyams, *Language Acquisition and the Theory of Parameters* , D. Reidel Publishing Company.

[**McLelland** *et al* **87**] D. MacLelland and J. Rumelhart, "Learning the Past Tense of English Verbs - Explicit Rules or Parallel Distributed Processing" in *Mechanisms of Language Acquisition* , edited by B. MacWhinney, Lawrence Erlbaum Assoc., Hillsdale, NJ.

[**Mills 84**] A. Mills, *The Acquisition of German in the Cross-Linguistic Study of Language Acquisition, Vol. 1* , D. Sloban ed., Lawrence Erlbaum Assoc., Hillsdale, NJ.

[**Newport** *et al.* **77**] E. Newport, H. Gleitman and L. Gleitman, "Mother, I'd Rather Do It Myself, Some Effects and Noneffects of Maternal Speech Style", in *Talking to Children: Language Input and Acquisition* , C. Snow and C. Ferguson, eds., Cambridge University Press, Cambridge.

[**Newport** *et al.* **84**] E. Newport, H. Gleitman and L. Gleitman, "The Current Status of the Motherese Hypothesis", *Journal of Child Language* 11: 43-80.

[**Pinker 79**] S. Pinker, "Formal Models of Language Learning", *Cognition* , 7: 217-283.

[**Pinker 87**] S. Pinker, "The Bootstrapping Problem in Language Acquisition", in *Mechanisms of Language Acquisition* , edited by B. MacWhinney, Lawrence Erlbaum Assoc., Hillsdale, NJ.

[**Wexler** *et al.* **87**] K. Wexler and M. R. Manzini, "Parameters and Learnability in Binding Theory", in T. Roeper and E. Williams, eds.

# Research Interests

Steven L. Lytinen
Carol E. Moon
Peter M. Hastings

Our research in machine learning of natural language has been centered around two issues:

1. **Second Language Acquisition.** In particular, we are interested in second language acquisition in the context of an instructional setting. We have designed a computer model of second language acquisition, called ANT, which learns grammar rules for a second language. It receives as input a set of lessons, each of which describes a grammar rule for the second language and provides a set of examples illustrating the use of the rule.

   We have begun to try to take our model seriously as a model of human second language acquisition. We have run a pilot study which supports some of our model's predictions, and we have plans to run more studies in the future to confirm other predictions about human second language acquisition. Our paper discusses the pilot study and some of our future plans.

2. **Inferring Word Meanings.** We are developing a program which can infer the meanings of unknown words over time from the context in which they appear. Our approach is incremental. As more examples are encountered in which a previously unknown word appears, the program refines its hypothesis as to the meaning of the unknown word. Our paper describes the algorithm which we have developed, and the assumptions that it is based on. The assumptions lead to many limitations to the algorithm, which are discussed in the paper.

## Relevant publications

Lytinen, S. and Moon, C. (1988). Learning a Second Language. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, St. Paul, MN, August 1988, pp. 222-226.

Moon, C. and Lytinen, S. (1989). The function of examples in learning a second language from an instructional text. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI, August 1989. Lawrence Erlbaum Associates, Hillsdale, NJ.

Lytinen, S. and Roberts, S. (1989). Lexical acquisition as a by-product of natural language processing. In *Proceedings of the First International Lexical Acquisition Workshop*, IJCAI-89, Detroit, MI, August 1989.

Lytinen, S. (1990) Robust processing of terse text. In *Proceedings of the 1990 AAAI Symposium on Intelligent Text-based Systems*, Stanford CA, March 1990, pp. 10-14.

Lytinen, S., and Moon, C. (1990). A comparison of learning techniques in second language learning. In *Proceedings of the Seventh International Conference on Machine Learning*, Austin, TX, June 1990,

# Cognitive Modeling of Second Language Acquisition

Steven L. Lytinen
Artificial Intelligence Laboratory
The University of Michigan
Ann Arbor, MI 48109

and

Carol E. Moon
Department of Computer Science
Harvey Mudd College
Claremont, CA 91711

## 1 Introduction

We are studying the acquisition of syntactic knowledge in second language learning. We have developed a computer model of second language acquisition, called ANT (Acquisition using Native-language Transfer) (Lytinen and Moon,1988; Moon and Lytinen, 1989; Lytinen and Moon, 1990; Moon, 1990). ANT successfully learns approximately 85% of the grammar rules presented in a typical first-year German textbook. Input to the system is similar to what is found in a typical introductory text, containing a mixture of instructions about a grammar rule and examples illustrating the rule. The system modifies its English grammar rules accordingly, so that they correspond to the grammar of German. ANT can then "understand" German sentences.

In developing ANT, we have focused on the issue of how instructions and examples interact with each other during the learning process. We have discovered several reasons why, from a functional standpoint, both instructions and examples are often useful in learning new grammar rules. From this work, we can make many specific predictions about second language learning performance. For example, we can characterize what types of rules are easy or hard for our program to learn, and factors which affect the difficulty of a rule, such as lesson format (i.e., the effectiveness of instructions and of examples for particular types of rules), and the order in which lessons are presented.

We have begun to test some of our model's predictions in a series of psychological experiments, to see if the they are valid for human second language learning. In particular, we have run a pilot study which tests some of our predictions about factors that determine the difficulty of different types of grammar rules, as well as the effects of lesson format on rule difficulty. The pilot study supports our predictions regarding these factors.

In this paper, we briefly describe the ANT model, and the predictions that the model makes regarding rule difficulty and lesson format. We then present the pilot study, and discuss future work on empirical testing.

## 2 The Current Model

To explain how ANT learns, we present an example lesson:

> In German, verbs come at the end of relative clauses.
> Examples:
>
> > Der Erdferkel, der Ameisen frißt, läuft langsam
> > (the aardvark who ants eats runs slowly)[1]
> > Der Mann, der mir Bücher gibt, wohnt in Paris.
> > (the man who me books gives lives in Paris)

ANT's task is to change its English grammar rules so that they will work for German relative clauses. Some of ANT's English relative clause rules are the following:[2]

(1) RC → RP VP
(2) RC → RP NP VP'

Although the instruction portion of the lesson describes a difference between German and English relative clauses, the modification to ANT's grammar does not involve its relative clause rules at all. This is because the verb is embedded in rules about verb phrases (VP and VP'). Some of these rules are the following:

(3) VP → V NP
(4) VP' → V[3]

Is is these rules which must be changed. Thus, the problem of finding the relevant English rules which must be modified is not an easy one. The instruction says something about verbs and relative clauses; it says nothing about verb phrases.

This is where examples come into play in the learning process. Without examples, ANT would have to search through its grammar for possible appearances of verbs within relative clauses. In the worst case, this could mean searching the entire grammar, since a verb could in theory appear inside of any constituent of a RC, and RC's could possibly contain every other kind of constituent. However, because ANT is provided with examples in addition to instructions, ANT parses the examples, letting them guide it to the rules which must be changed. During the parse, ANT is forced to use the rules which must be modified for German. Thus, the potentially large search through the grammar is avoided.

Because the instruction portion of our example lesson tells ANT that it is learning a change in word order within relative clauses, the ordering constraints in its relative clause rules are relaxed when parsing the examples. As a result, it is able to parse a sentence whose relative clause word ordering does not conform to English grammar. Let us consider the first example from above. ANT parses it. The relevant portion of the parse tree which is produced is shown in figure 1. The new form of of rule (3) above is extracted from the example's parse tree, producing the following new German rule:

(3') VP → NP V

[1] ANT does not receive an English translation as part of its input. The literal English translation is provided here for the benefit of the reader.

[2] Although ANT's linguistic knowledge is encoded in a unification-style grammar (Shieber, 1986), for our purposes here we can assume that they are context-free rules.

[3] The unification form of these rules enforces the verb type appropriately; for example, only transitive verbs may appear in the constructions specified by rules 3 and 4.

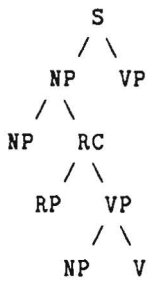



Figure 1: Parse tree produced by ANT from German example

This rule is overly general, though, since it should only hold for VP's within relative (and other subordinate) clauses. ANT avoids this mistake by creating a new category (CL-VP). It knows to do this because of the information from the instruction. The final rules, then, are:

(1') RC → RP CL-VP
(2) RC → RP NP VP'
(3') CL-VP → NP V
(4) VP' → V

After several examples illustrating the relative clause construction for other types of verb phrases, ANT successfully modifies all of its original English rules in a similar fashion.

# 3 Empirical Testing

ANT's performance on the relative clause rule suggests (at least) two factors that might play a role in determining how difficult it to learn a grammar rule. These factors are:

1. **Ease of access of relevant native language knowledge.** For ANT, the most difficult part of learning the German relative clause rule is knowing which English rules to change. Identifying the relevant English rules is due to the "embeddedness" of the change: verbs are embedded inside of relative clauses in rules about relative clause constituents. Thus, our system would have an easier time with rule changes that are less embedded. For example, the Spanish rule that direct object pronouns precede the verb is easier for ANT to learn, because a verb and its direct object appear in the same grammar rule, VP → V NP.
2. **Effects of lesson format.** Both instructions and examples play crucial roles in ANT's learning process. Thus, ANT's performance would be adversely affected if either were missing from a lesson. However, this affect depends in part on rule difficulty: examples play a crucial role in difficult rules (by our embeddedness criterion), but for easy rules (non-embedded changes), an instructions-only lesson format should not cause as much trouble.

We conducted a pilot study to test whether these two factors are important in human second language acquisition. In the study, we began to explore both the difficulty factor and lesson format factor on learning performance. 12 subjects were presented with lessons which taught them a grammar rule for a hypothetical variant of English. Lessons varied in two ways: in the difficulty of the rule being taught, and in the format of the lesson. Rule difficulty was based on our embeddedness criterion discussed earlier. Thus, rules with embedded changes were predicted to be more difficult to learn than those with unembedded

| | Error rates | | | Timings (secs.) | | |
|---|---|---|---|---|---|---|
| | Instrs Only | Examples Only | Instrs & Examples | Instrs Only | Examples Only | Instrs & Examples |
| Difficult | .42 | .21 | .25 | 15.5 | 13.6 | 11.0 |
| Easier | .25 | .04 | .04 | 12.3 | 10.4 | 7.8 |

Figure 2: Results of translation task for different rule types and lesson formats

changes. Lessons were in one of three formats. One third of the lessons each subject saw were in an instruction-only format, consisting of a short description of the grammar rule, but no examples. One third of the lessons consisted only of a set of examples illustrating the rule to be learned. Finally, one third of the lessons consisted of both a description of the grammar rule and a set of examples. Each subject received two lessons of each possible combination of rule type and lesson format: difficult vs. hard rules, and instructions only, examples only, and mixed format. Rules were presented in each different lesson format to equal numbers of subjects.

After presentation of each lesson, subjects were then given a sentence in "normal" English and asked to produce a spoken translation of it in the variant dialect, using the rule they had just learned. Their performance was measured in terms of error rate and production time. Errors were only counted if they involved misapplication of the newly learned rule.

We predicted that error rates and timings would be higher for difficult than for easy rules, according to the difficulty criteria from our computer model. We also predicted that, in general, mixed format lessons would facilitate learning better than either instructions or examples alone. Finally, we predicted an interaction between the two variables. Based on our model, we would predict that the inclusion of examples along with instructions in the lesson should facilitate learning more for difficult rules than for easy ones.

The results of the study are summarized in figure 2. First, error rates for difficult rules were significantly higher than for easy rules, according to our difficulty criteria ($F(1,11)=6.22$; $p=.03$). Error rates were also higher for rules learned from instructions only than for those learned from instructions and examples ($F(2,22)=5.41$; $p=.012$). Timing results also showed a significant difficulty effect ($F(1,11)=13.78$; $p<.005$). Lesson format effects were in the right direction, but were not significant for the timings.

These results support our characterization of one factor that influences the difficulty of a new grammar rule. They also indicate that in general, examples presented in a lesson either alone or with instructions expedite the learning process as compared to instructions-only lessons.

# 4 Future Work

Though the pilot study confirms some of our predictions, we discovered some possible methodological concerns to be addressed in further experimentation. First, the translation task could have an effect on likelihood of transfer, thus affecting the likelihood of error in performance. We plan to use several tasks in the full study, including grammaticality judgement of several tasks should guarantee that the results are not biased by the peculiarities of a single task.

Second, we plan to alter our style of presentation of lessons. In the pilot study, lessons were displayed for an initial learning period, then were available for inspection during the translation task. This seemed to negatively affect the amount of effort that subjects put into learning the rule before being asked to perform the translation task. Since the lesson was available during the task, subjects seemed to put off learning the rule until it was required during the translation. This probably affected performance, certainly in terms of translation times, and possibly in terms of number of errors produced. In particular,

it could explain why subjects performed as well with examples-only lessons as they did with mixed format lessons. In addition, forcing subjects to learn and remember rules is more similar to instruction and use in natural settings.

Finally, the pilot study was rather limited in the range of grammar rules that were used. We plan to expand the range of rule types, testing several criteria for rule difficulty.

We wish to explore other issues in a similar manner. One such issue is lesson sequence. Our model also can be used to make predictions about effects of different sequencings of lessons on learning. Depending on what has already been taught, a new rule can be relatively harder or easier to learn. For example, learning correct German verb location involves learning two new rules for English speakers: the verb is placed in the second position in main clauses, but at the end in subordinate clauses. Our model predicts that learning the general rule of the verb coming second in main clauses before learning the location of verbs in subordinate clauses should facilitate learning more easily. This is because there is less question about which rule takes precedence if the more general rule is learned first: it is easier to learn a special case which overrides the general rule if the special case is learned after the general rule (according to our model). We propose to vary the order in which sequences of grammar rules is learned, to verify the accuracy of our predictions.

In addition, we are exploring further refinements of our computer model. One important issue is the degree to which transfer is utilized in ANT. Our current model makes strong assumptions about transfer. Specifically, ANT uses corresponding English grammar rules as a starting point for constructing its German rules whenever possible. Although there is psycholinguistic evidence for transfer (e.g., Jansen, Lalleman, and Muysken, 1981; Snow, 1981; Selinker, 1969), other evidence indicates that people do not always transfer native language knowledge to a foreign language (e.g., Rutherford, 1983). We are exploring this phenomenon, to try to characterize further the situations in which transfer does and does not occur. We wish to model more closely the data from existing studies, which suggests conditions that affect transfer.

## References


Jansen, B., Lelleman, J., and Muysken, P. (1981). The alternation hypothesis: Acquisition of Dutch word order by Turkish and Moroccan foreign workers. *Language Learning, 31*, pp. 315-316.

Lytinen, S., and Moon, C. (1988). Learning a second language. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, St. Paul, MN, pp. 222-226.

Moon, C., and Lytinen, S. (1989). The function of examples in learning a second language from an instructional text. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI. Lawrence Erlbaum Associates, Hillsdale, NJ.

Lytinen, S., and Moon, C. (1990). A comparison of learning techniques in second language learning. In *Proceedings of the Seventh International Conference on Machine Learning*, Austin, TX, pp. 377-383.

Rutherford, W. (1983). Language typology and language transfer. In Gass, S., and Selinker, L. (eds.), *Language Transfer in Language Learning.* Newbury House, Rowley, MA.

Selinker, L. (1969). Language transfer. *General Linguistics 9*, pp. 67-92.

Shieber, S. (1986). *An Introduction to Unification-based Approaches to Grammar.* CSLI, Stanford CA.

Snow, C. (1981). English speakers' acquisition of Dutch syntax. In Winitz, H. (ed.), *Annals of the New York Academy of Sciences.* New York Academy of Sciences, New York.

# Distituent Parsing and Grammar Induction

David M. Magerman
CS Department
Stanford University

The work I will be presenting was not originally intended to address the problem of grammar learning directly. The distituent parser was meant be a natural language parser for unrestricted text that didn't have the computational expense of a grammar-based parser and didn't require a hand-written grammar. However, since I was trying to determine the syntactic structure of examples from a language without specifying the language's grammar, I was in essence trying to learn grammar.

The mutual information distituent parser was my undergraduate senior thesis, and Mitch Marcus was my thesis advisor and co-author. I am currently a first-year graduate student in computer science at Stanford. Aside from this project, I have also developed a probabilistic parser, Pearl, which estimates context-sensitive conditional probabilities of grammar rules in order to learn attachment tendencies from a corpus of examples. In my last two years at the University of Pennsylvania, I also assisted Mitch Marcus on the Penn Treebank project.

## References

[1] "Parsing the Voyager Corpus with Pearl," David M. Magerman and Mitchell P. Marcus, in *Proceedings, DARPA Speech and Natural Language Workshop,* February, 1991.

[2] "Pearl: A Probabilistic Chart Parser," David M. Magerman and Mitchell P. Marcus, in *Proceedings, European ACL,* April, 1991 and in *Proceedings, Second International Workshop for Parsing Technologies,* February, 1991.

[3] "Parsing a Natural Language Using Mutual Information Statistics," David M. Magerman and Mitchell P. Marcus, in *Proceedings, Eight National Conference on Artificial Intelligence,* August, 1990.

[4] "Deducing Linguistic Structure from Very Large Corpora," Eric Brill, David M. Magerman, Mitchell P. Marcus, and Beatrice Santorini, in *Proceedings, Speech and Natural Language Workshop,* June, 1990.

**Abstract**

The purpose of this paper is to characterize a constituent boundary parsing algorithm, using an information-theoretic measure called generalized mutual information, which serves as an alternative to traditional grammar-based parsing methods. This method is based on the hypothesis that constituent boundaries can be extracted from a given sentence (or word sequence) by analyzing the mutual information values of the part-of-speech $n$-grams within the sentence. This hypothesis is supported by the performance of an implementation of this parsing algorithm which determines a recursive unlabeled bracketing of unrestricted English text with a relatively low error rate. By using the constituents from the distituent parser, noun phrase and preposition phrase categories can be induced from a corpus. While the error rate is still to high to allow for grammar induction, we present a method for reducing this error rate by enforcing a simple linguistic assumption on the parser.

# Distituent Parsing and Grammar Induction*

David M. Magerman and Mitchell P. Marcus
CIS Department
University of Pennsylvania
Philadelphia, PA 19104
E-mail: magerman@linc.cis.upenn.edu

## Introduction

A standard approach to parsing a natural language is to characterize the language using a set of rules, a grammar. A grammar-based parsing algorithm recursively determines a sequence of applications of these rules which reduces the sentence to a single category. Besides determining sentence structure, grammar-based approaches can also identify attributes of phrases, such as case, tense, and number, and they are known to be extremely effective at characterizing and classifying sentences. But these techniques are generally demonstrated using only a subset of the grammar of the language. In order for a grammar-based parser to be applied to unrestricted natural language text, it must account for most of the complexities of the natural language. Thus, one must first concisely describe the bulk of the grammar of that language, an extremely difficult task.

This characterization suggests that a solution to the problem of parsing unrestricted natural language text must rely on an alternative to the grammar-based approach. The approach presented in this paper is based on viewing part-of-speech sequences as stochastic events and applying probabilistic models to these events. Our hypothesis is that constituent boundaries, or "distituents," can be extracted from a sequence of $n$ categories, or an $n$-gram, by analyzing the mutual information values of the part-of-speech sequences within that $n$-gram. In particular, we will demonstrate that the generalized mutual information statistic, an extension of the bigram (pairwise) mutual information of two events into $n$-space, acts as a viable measure of continuity in a sentence.

One notable attribute of our algorithm is that it actually includes a grammar — a distituent grammar, to be precise. A distituent grammar is a list of tag pairs which *cannot* be adjacent within a constituent. For instance, *noun prep* is a known distituent in English, since the grammar of English does not allow a constituent consisting of a noun followed by a preposition. Notice that the nominal head of a noun phrase may be followed by a prepositional phrase; in the context of distituent parsing, once a sequence of tags, such as *(prep noun)*, is grouped as a constituent, it is considered as a unit.

Based on our claim, mutual information *should* detect distituents without aid, and a distituent grammar should not be necessary. However, the application of mutual information to natural language parsing depends on a crucial assumption about constituents in a natural language. Given any constituent $n$-gram, $a_1 a_2 \ldots a_n$, the probability of that constituent occurring is usually significantly higher than the probability of $a_1 a_2 \ldots a_n a_{n+1}$ occurring. This is true, in general, because most constituents appear in a variety of contexts. Once a constituent is detected, it is usually very difficult to predict what part-of-speech will come next. While this assumption is not valid in every case, it turns out that a handful of cases in which it is invalid are responsible for a majority of the errors made by the parser. It is in these few cases that we appeal to the distituent grammar to prevent these errors.

The distituent parsing algorithm is an example of a stochastic, corpus-based approach to parsing. In the past, a significant disadvantage of probabilistic parsing techniques has been that these methods were prone to higher than acceptable error rates. By contrast, the mutual information parsing method presented in this paper is based on a statistic which is both highly accurate and, in the cases where it is inaccurate, highly consistent. Taking advantage of these two attributes, the generalized mutual information statistic and the distituent grammar combine to parse sentences with, on average, two errors per sentence for sentences of 15 words or less, and five errors per sentence for sentences of 30 words or less (based on sentences from a reserved test subset of the Tagged Brown Corpus, see footnote 1). Many of the errors on longer sentences result from conjunctions, which are traditionally troublesome for grammar-based algorithms as well. Further,

*This work was partially supported by DARPA grant No. N0014-85-K0018, by DARPA and AFOSR jointly under grant No. AFOSR-90-0066, and by ARO grant No. DAAL 03-89-C0031 PRI. Special thanks to Ken Church, Stuart Shieber, Max Mintz, Beatrice Santorini, and Tom Veatch for their valued input, guidance and support.

this parsing technique is extremely efficient, parsing a 35,000 word corpus in under 10 minutes on a Sun 4/280.

It should be noted at this point that, while many stochastic approaches to natural language processing that utilize frequencies to estimate probabilities suffer from sparse data, *sparse data is not a concern in the domain of our algorithm.* Sparse data usually results from the infrequency of *word* sequences in a corpus. The statistics extracted from our training corpus are based on tag $n$-grams for a set of 64 tags, not word $n$-grams.[1] The corpus size is sufficiently large that enough tag $n$-grams occur with sufficient frequency to permit accurate estimates of their probabilities. Therefore, the kinds of estimation methods of $(n+1)$-gram probabilities using $n$-gram probabilities discussed in Katz (1987) and Church & Gale (1989) are not needed.

This line of research was motivated by a series of successful applications of mutual information statistics to other problems in natural language processing. In the last decade, research in speech recognition (Jelinek 1985), noun classification (Hindle 1988), predicate argument relations (Church & Hanks 1989), and other areas have shown that mutual information statistics provide a wealth of information for solving these problems.

## Mutual Information Statistics

Before discussing the mutual information parsing algorithm, we will demonstrate the mathematical basis for using mutual information statistics to locate constituent boundaries. Terminology becomes very important at this point, since there are actually two statistics which are associated with the term "mutual information," the second being an extension of the first.

In his treatise on information theory, *Transmission of Information* (Fano 1961), Fano discusses the mutual information statistic as a measure of the interdependence of two signals in a message. This bigram mutual information is a function of the probabilities of the two events:

$$\mathcal{MI}(x,y) = \log \frac{\mathcal{P}_{X,Y}(x,y)}{\mathcal{P}_X(x)\mathcal{P}_Y(y)}. \quad (1)$$

Consider these events not as signals but as parts-of-speech in sequence in a sentence. Then an estimate of the mutual information of two categories, $xy$, is:

$$\mathcal{MI}(x,y) \approx \log \frac{\frac{\#\ xy \text{ in corpus}}{\text{total \# of bigrams in corpus}}}{\left(\frac{\#\ x}{\text{corpus size}}\right)\left(\frac{\#\ y}{\text{corpus size}}\right)}. \quad (2)$$

[1]The corpus we use to train our parser is the Tagged Brown Corpus (Francis and Kucera, 1982). Ninety percent of the corpus is used for training the parser, and the other ten percent is used for testing. The tag set used is a subset of the Brown Corpus tag set.

In order to take advantage of context in determining distituents in a sentence, however, one cannot restrict oneself to looking at pairs of tokens, or bigrams; one must be able to consider $n$-grams as well, where $n$ spans more than one constituent. To satisfy this condition, we can simply extend mutual information from bigrams to $n$-grams by allowing the events $x$ and $y$ to be part-of-speech $n$-grams instead of single parts-of-speech. We will show that this extension is not sufficient for the task at hand.

The second statistic associated with mutual information is what we will call "generalized mutual information," because it is a generalization of the mutual information of part-of-speech bigrams into $n$-space. Generalized mutual information uses the context on both sides of adjacent parts-of-speech to determine a measure of its distituency in a given sentence. We will discuss this measure below.

While our distituent parsing technique relies on generalized mutual information of $n$-grams, the foundations of the technique will be illustrated with the base case of simple mutual information over the space of bigrams for expository convenience.

## Mutual Information

The bigram mutual information of two events is a measure of the interdependence of these events in sequence. In applying the concept of mutual information to the analysis of sentences, we are concerned with more than just the interdependence of a bigram. In order to take into account the context of the bigram, the interdependence of part-of-speech $n$-grams (sequences of $n$ parts-of-speech) must be considered. Thus, we consider an $n$-gram as a bigram of an $n_1$-gram and an $n_2$-gram, where $n_1 + n_2 = n$. The mutual information of this bigram is

$$\mathcal{MI}(n_1\text{-gram}, n_2\text{-gram}) = \log \frac{\mathcal{P}[n\text{-gram}]}{\mathcal{P}[n_1\text{-gram}]\mathcal{P}[n_2\text{-gram}]}. \quad (3)$$

Notice that there are $(n-1)$ ways of partitioning an $n$-gram. Thus, for each $n$-gram, there is an $(n-1)$ vector of mutual information values. For a given $n$-gram $x_1 \ldots x_n$, we can define the mutual information values of $x$ by:

$$\mathcal{MI}_n^k(x_1 \ldots x_n) = \mathcal{MI}(x_1 \ldots x_k, x_{k+1} \ldots x_n) \quad (4)$$

$$= \log \frac{\mathcal{P}(x_1 \ldots x_n)}{\mathcal{P}(x_1 \ldots x_k)\mathcal{P}(x_{k+1} \ldots x_n)} \quad (5)$$

where $1 \leq k < n$.

Notice that, in the above equation, for each $\mathcal{MI}_n^k(x)$, the numerator, $\mathcal{P}(x_1 \ldots x_n)$, remains the same while the denominator, $\mathcal{P}(x_1 \ldots x_k)\mathcal{P}(x_{k+1} \ldots x_n)$, depends on $k$. Thus, the mutual information value achieves its minimum at the point where the denominator is maximized. The empirical claim to be tested in this paper is that the minimum is achieved when the two components of this $n$-gram are in two different constituents,

i.e. when $x_k x_{k+1}$ is a distituent. Our experiments show that this claim is largely true with a few interesting exceptions.

The motivation for this claim comes from examining the characteristics of $n$-grams which contain pairs of constituents. Consider a tag sequence, $x_1 \ldots x_n$, which is composed of two constituents $x_1 \ldots x_k$ and $x_{k+1} \ldots x_n$. Since $x_1 \ldots x_k$ is a constituent, $x_1 \ldots x_{k-1}$ is very likely to be followed by $x_k$. Thus,

$$\mathcal{P}(x_1 \ldots x_k) \approx \mathcal{P}(x_1 \ldots x_{k-1}). \quad (6)$$

By the same logic,

$$\mathcal{P}(x_{k+1} \ldots x_n) \approx \mathcal{P}(x_{k+2} \ldots x_n). \quad (7)$$

On the other hand, assuming $x_k$ and $x_{k+1}$ are uncorrelated (in the general case),

$$\mathcal{P}(x_k \ldots x_n) \ll \mathcal{P}(x_{k+1} \ldots x_n) \quad (8)$$

and

$$\mathcal{P}(x_1 \ldots x_{k+1}) \ll \mathcal{P}(x_1 \ldots x_k). \quad (9)$$

Therefore,

$$\begin{aligned} &\mathcal{MI}(x_1 \ldots x_k, x_{k+1} \ldots x_n) \\ &= \log \frac{\mathcal{P}(x_1 \ldots x_n)}{\mathcal{P}(x_1 \ldots x_k)\mathcal{P}(x_{k+1} \ldots x_n)} && (10) \\ &\approx \log \frac{\mathcal{P}(x_1 \ldots x_n)}{\mathcal{P}(x_1 \ldots x_{k-1})\mathcal{P}(x_{k+1} \ldots x_n)} && (11) \\ &> \log \frac{\mathcal{P}(x_1 \ldots x_n)}{\mathcal{P}(x_1 \ldots x_{k-1})\mathcal{P}(x_k \ldots x_n)} && (12) \\ &= \mathcal{MI}(x_1 \ldots x_{k-1}, x_k \ldots x_n). && (13) \end{aligned}$$

By applying a symmetry argument and using induction, the above logic suggests the hypothesis that, in the general case, if a distituent exists in an $n$-gram, it should be found where the minimum value of the mutual information vector occurs.

There is no significance to the individual mutual information values of an $n$-gram other than the minimum; however, the distribution of the values is significant. If all the values are very close together, then, while the most likely location of the distituent is still where the minimum occurs, the confidence associated with this selection is low. Conversely, if these values are distributed over a large range, and the minimum is much lower than the maximum, then the confidence is much higher that there is a distituent where the minimum occurs. Thus, the standard deviation of the mutual information values of an $n$-gram is an estimate of the confidence of the selected distituent.

## Generalized Mutual Information

Although bigram mutual information can be extended simply to $n$-space by the technique described in the previous section, this extension does not satisfy the needs of a distituent parser. A distituent parsing technique attempts to select the most likely distituents based on its statistic. Thus, a straightforward approach would assign each potential distituent a single real number corresponding to the extent to which its context suggests it is a distituent. But the simple extension of bigram mutual information assigns each potential distituent a number for each $n$-gram of which it is a part. The question remains how to combine these numbers in order to achieve a valid measure of distituency.

Our investigations revealed that a useful way to combine mutual information values is, for each possible distituent $xy$, to take a weighted sum of the mutual information values of all possible pairings of $n$-grams ending with $x$ and $n$-grams beginning with $y$, within a fixed size window. So, for a window of size $w = 4$, given the context $x_1 x_2 x_3 x_4$, the generalized mutual information of $x_2 x_3$ :

$$\begin{aligned} &\mathcal{GMI}_4(x_1 x_2, x_3 x_4), \\ &\quad = k_1 \mathcal{MI}(x_2, x_3) + k_2 \mathcal{MI}(x_2, x_3 x_4) + && (14) \\ &\quad\quad k_3 \mathcal{MI}(x_1 x_2, x_3) + k_4 \mathcal{MI}(x_1 x_2, x_3 x_4) && (15) \end{aligned}$$

which is equivalent to

$$\log \left( k \frac{\mathcal{P}[x_2 x_3]\mathcal{P}[x_2 x_3 x_4]\mathcal{P}[x_1 x_2 x_3]\mathcal{P}[x_1 x_2 x_3 x_4]}{[\mathcal{P}[x_2]\mathcal{P}[x_3]\mathcal{P}[x_1 x_2]\mathcal{P}[x_3 x_4]]^2} \right) \quad (16)$$

In general, the generalized mutual information of any given bigram $xy$ in the context $x_1 \ldots x_{i-1} x y y_1 \ldots y_{j-1}$ is equivalent to

$$\log \left( \frac{\prod_{X \text{ crosses } xy} k_X \mathcal{P}[X]}{\prod_{X \text{ does not cross } xy} \mathcal{P}[X]^{(i+j)/2}} \right). \quad (17)$$

This formula behaves in a manner consistent with one's expectation of a generalized mutual information statistic. It incorporates all of the mutual information data within the given window in a symmetric manner. Since it is the sum of bigram mutual information values, its behavior parallels that of bigram mutual information.

The weighting function which should be used for each term in the equation was alluded to earlier. The standard deviation of the values of the bigram mutual information vector of an $n$-gram is a valid measure of the confidence of these values. Since distituency is indicated by mutual information minima, the weighting function should be the reciprocal of the standard deviation.

In summary, the generalized mutual information statistic is defined to be:

$$\begin{aligned} &\mathcal{GMI}_{(i+j)}(x_1 \ldots x_i, y_1 \ldots y_j) \\ &\quad = \sum_{\substack{X \text{ ends with } x, \\ Y \text{ begins with } y_1}} \frac{1}{\sigma_{XY}} \mathcal{MI}(X, Y), && (18) \end{aligned}$$

where $\sigma_{XY}$ is the standard deviation of the $\mathcal{MI}^k_{|XY|}$ values within $XY$.

## The Parsing Algorithm

Due to space limitations, I will forego a detailed description of the parsing algorithm here. For more information about the parsing algorithm, see "Parsing a Natural Language Using Mutual Information Statistics" in the proceedings of AAAI-90.

## Distituent Parsing and Grammar Induction

Although we have only done preliminary work on grammar induction using this distituent parsing technique, the results we have obtained suggest that distituent parsing may be useful in the initial phases of grammar learning.

The initial experiment we performed involved parsing about 35,000 words of text randomly selected from the Brown Corpus. By examining the distributions of the contexts of the constituents which the parser discovered, and clustering constituents which have similar distributions, we were able to induce most of the noun phrase and prepositional phrase categories which occurred in the corpus.

However, because the parser makes some systematic errors, there was also a lot of noise generated by this experiment. The frequency and consistency of the parser errors make it very difficult to distinguish between linguistically valid constituents and incorrect structures. Thus, other than the noun phrase and prepositional phrase categories, the rest of the categories discovered were filled with errors, and were generally unuseful.

In order to make grammar induction via distituent parsing more feasible, we must filter out the errors made by the distituent parser. One way to eliminate the errors is to make linguistic assumptions about language. For instance, we could assume that every constituent has a head and that a head must be either initial or final. Given this information, we could parse the language once, determine from the constituents discovered what the possible head categories for the language are, and reparse the language enforcing the head principle. Although we have not yet undertaken this experiment, based on the types of errors made by the parser, we believe this technique will be effective on languages for which this linguistic assumption is true.

## Conclusion

We have presented parsing technique which serves as an alternative to traditional grammar-based parsing. By searching for constituent boundaries, or distituents, instead of fully-specified constituents, distituent parsing eliminates the need for bulky grammars, and provides a computationally feasible method for determining syntactic sentence structure.

Since distituent parsing can be accomplished by training a statistical measure from an unparsed corpus, it may serve as a viable method for inducing grammars for natural languages. Although the error rate of the mutual information-based distituent parser is currently too high for inducing all of the constituent classes from a corpus, we may be able to reduce this error rate significantly by making universal linguistic assumptions about language.

## References


[1] Church, K. 1988. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In Proceedings of the Second Conference on Applied Natural Language Processing. Austin, Texas.

[2] Church, K.; and Gale, W. 1990. Enhanced Good-Turing and Cat-Cal: Two New Methods for Estimating Probabilities of English Bigrams. *Computers, Speech and Language.*

[3] Church, K.; and Hanks, P. 1989. Word Association Norms, Mutual Information, and Lexicography. In Proceedings of the 27th Annual Conference of the Association of Computational Linguistics.

[4] Fano, R. 1961. *Transmission of Information.* New York, New York: MIT Press.

[5] Francis, W.; and Kucera, H. 1982. *Frequency Analysis of English Usage: Lexicon and Grammar.* Boston, Mass.: Houghton Mifflin Company.

[6] Hindle, D. 1988. Acquiring a Noun Classification from Predicate-Argument Structures. Bell Laboratories.

[7] Jelinek, F. 1985. Self-organizing Language Modeling for Speech Recognition. IBM Report.

[8] Katz, S. M. 1987. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-35, No. 3.*

# The Automatic Acquisition of Linguistic Structure from Large Corpora: An Overview of Work at the University of Pennsylvania

**Mitchell Marcus**
Department of Computer and Information Science
University of Pennsylvania
(email: mitch@cis.upenn.edu)

The past five years have seen the beginning of a major shift of research focus in natural language processing. After twenty years of primary emphasis on online systems which crucially depend upon the magical ability of users to adapt to the limitations of the system, a new generation of systems is emerging that both extract information from and summarize pre-existing text from real-world domains. To achieve high coverage in such systems, a wide variety of research breakthroughs will be necessary. One advance which is critical to truly robust wide-coverage systems is a technology which allows the automatic acquisition of linguistic structure through the analysis of both literal and annotated text corpora. Research results already in hand suggest that significant progress in this area, at least in the area of syntax, may occur in the next few years.

We at Penn have initiated a research program to see how far the paradigm of trainable systems can take us towards the fully automatic analysis of unconstrained text. We are proceeding under the assumption that this work should proceed by attempting to combine two different traditions often viewed as mutually exclusive: the research program of generative grammar, as set forth originally by Noam Chomsky and the research paradigm of distributional analysis, as developed by the American structural linguists resulting in the work of Zellig Harris [4].

## Information Theoretic Parsing

This investigation of distributional analysis has already yielded results which are both surprising and encouraging. We have investigated how accurately the grammatical structure of a sentence can be determined without an explicitly encoded grammar at all, using only automatically compiled distributional statistics of a corpus of text which has been hand tagged for part of speech.

As part of this research, we have developed a constituent boundary parsing algorithm which derives an (unlabelled) bracketing given text annotated for part of speech as input [5]. This method is based on the hypothesis that constituent boundaries can be extracted from a given part-of-speech $n$-gram by analyzing the mutual information values within the $n$-gram, extended to a new generalization of the information theoretic measure of *mutual information*. This hypothesis is supported by the performance of an implementation of this parsing algorithm which determines recursively nested sentence structure, with an error rate of roughly 2 misplaced boundaries for test sentences of length 10-15 words, and five misplaced boundaries for sentences of 15-30 tokens. (All test sentences were randomly selected from a reserved test corpus.) We discuss below a mechanism to deal with the limited set of specific circumstances in which the hypothesis fails.

The mutual information statistic [2] is a measure of the interdependence of two signals in a message. It is a function of the probabilities of the two events:

$$\mathcal{MI}(x,y) = \log \frac{\mathcal{P}_{X,Y}(x,y)}{\mathcal{P}_X(x)\mathcal{P}_Y(y)}.$$

In this paper, the events $x$ and $y$ will be not single parts-of-speech, but part-of-speech $n$-grams.

This work proceeds by viewing the part-of-speech sequences that make up sentences as stochastic events and applying probabilistic models to these events. It tests the hypothesis that constituent boundaries, or "distituents," can be extracted from a sequence of $n$ categories, or an $n$-gram, by analyzing the mutual information values of the part-of-speech sequences within that $n$-gram. More particularly, this hypothesis assumes that, given any constituent $n$-gram, $a_1 a_2 \ldots a_n$, the probability of that constituent occurring is usually significantly higher than the probability of $a_1 a_2 \ldots a_n a_{n+1}$ occurring.

The performance of the new algorithm demonstrates that the generalized mutual information statistic, an extension of the bigram (pairwise) mutual information of two events into $n$-space, acts as a viable measure of continuity in a sentence. This is true, in general, because most constituents appear in a variety of contexts. Once a constituent is detected, it is usually very difficult to predict what part-of-speech will come next. As it turns out, however, there are cases in which this assumption is not valid, but only a handful of these cases are responsible for a majority of the errors made

by the parser. To deal with these cases, our algorithm includes what we will call a distituent grammar — a list of tag pairs which *cannot* be adjacent within a constituent. One such pair is *noun prep*, since English does not allow a constituent consisting of a noun followed by a preposition. Notice that the nominal head of a noun phrase may be followed by a prepositional phrase; in the context of distituent parsing, once a sequence of tags, such as *(prep noun)*, is grouped as a constituent, it is considered as a unit. Our current distituent grammar consists of four rules of two tokens each.

Our current implementation of this parsing algorithm determines a recursive unlabeled bracketing of unrestricted English text. As stated above, the generalized mutual information statistic and the distituent grammar combine to parse sentences with, on average, two errors per sentence for sentences of 15 words or less, and five errors per sentence for sentences of 30 words or less (based on sentences from a reserved test subset of the Tagged Brown Corpus). Many of the errors on longer sentences result from conjunctions, which are traditionally troublesome for grammar-based algorithms as well. Further, this parsing technique is reasonably efficient, parsing a 35,000 word corpus in under 10 minutes on a Sun 4/280.

## Determining lexical features and part of speech

To allow this technique to be applied to completely unannotated text, we are concurrently experimenting with techniques to automatically derive the feature set and word classes of a language.[1] from a large corpus of text, again using only distributional facts. These techniques are based upon the following idea, a variant of the distributional analysis methods from Structural Linguistics ([3], [4]): features license the distributional behavior of lexical items. At the two extremes, a word with no features would not be licensed to appear in any context at all, whereas a word marked with all features of the language would be licensed to appear in every possible context.

The feature discovery system works as follows. First, a large amount of text is examined to discover the frequency of occurrence of different bigrams.[2] Based upon this data, the system groups words into classes. Two words are in the same class if they can occur in the same contexts. In order to determine whether **x** and **y** belong to the same class, the sytem first examines all bigrams containing **x**. If for a high percentage of these bigrams, the corresponding bigram with **y** substituted for **x** exists in the corpus, then it is likely that **y** has all of the features that **x** has (and maybe more). If upon examining the bigrams containing **y** the system is able to conclude that **x** also has all of the features that **y** has, it then concludes that **x** and **y** are in the same class.

For every pair of bigrams, the system must determine how much to weigh the presence of those bigrams as evidence that two words have features in common. For instance, assume: (a) the bigram *the boy* appears many times in the corpus being analyzed, while *the sits* never occurs. Also assume: (b) the bigram *boy the* (as in *the* **boy the** *girl kissed* ...) occurs once and *sits the* never occurs. Case (a) should be much stronger evidence that **boy** and **sits** are not in the same class than case (b). For each bigram $\alpha$**x** occurring in the corpus, evidence offered by the presence (or absence) of the bigram $\alpha$**y** is scaled by the frequency of $\alpha$**x** in the text divided by the total number of bigrams containing **x** on their right hand side. Since the end-of-phrase position is less restrictive, we would expect each bigram involving this position and the word to the right of it to occur less frequently than bigrams of two phrase-internal words. By weighing the evidence, bigrams which cross boundaries will be weighed less than those which do not. See [1] for more information and some preliminary results.

## Verb acquisition

We are also developing a computational model of verb acquisition which uses what we will call the principle of *structured overcommitment* (a specialization of the subset principle) to eliminate the need for negative evidence. The learner escapes from the need to be told that certain possibilities cannot occur (i.e. are"ungrammatical") by one simple expedient: It assumes that all properties it has observed are either obligatory or forbidden until it sees otherwise, at which point it decides that what it thought was either obligatory or forbidden is merely optional. This model is built upon a classification of verbs based upon a simple three-valued set of features which represents key aspects of a verb's syntactic structure, its predicate/argument structure, and the mapping between them. This model was originally implemented and tested working with a small set of hand-selected examples (see [7]); we hope to extend this work using large natural corpora in the near future.

We are also using the techniques discussed above to determine verb classes using $n$-gram techniques. We have been able to show (counter to any reasonable expectation) that a purely local examination of the two words (one to the right and one to the left) that occur immediately adjacent to a given verb provides enough information to hierarchically cluster these verbs into meaningful and fairly fine-grained grammatical categories, even distinguishing benefactive verbs (verbs that take an indirect object, roughly) into verbs of propositional attitude (e.g. *tell*) from verbs of physical transfer (e.g. *give*).

[1] We consider the set of features of a particular language to be all attributes which that language makes reference to in its syntax.

[2] For this experiment, we take a very local view of context, only considering bigrams.

## Probabilistic CF Parsing

In another experiment, in collaboration with UNISYS, we have investigated how distributional facts can be used to choose between the multiple grammatically acceptable analyses of a single sentence. We have developed (see [6]) a natural language parsing algorithm for unrestricted text which uses a novel probability-based scoring function to select the "best" parse of a sentence. The parser, Pearl, is a time-asynchronous bottom-up chart parser with Earley-type top-down prediction which pursues the highest-scoring theory in the chart, where the score of a theory represents the extent to which the context of the sentence predicts that interpretation. This parser differs from previous attempts at stochastic parsers in that it uses a richer form of conditional probabilities based on context to predict likelihood. In preliminary tests, Pearl has shown promising results in handling part-of-speech assignment, prepositional phrase attachment, and unknown word categorization. Trained on a corpus of 1100 sentences from MIT's Voyager direction-finding system and using the string grammar from UNISYS' PUNDIT Language Understanding System, Pearl correctly parsed 35 out of 40 or 88% of test sentences from previously unseen Voyager sentences.

## The Penn Treebank Project

To faciliate the kind of statistical experiments discussed above, both by us and by researchers at other institutions, we have undertaken the development of a a large annotated corpus of American English, annotated both with part-of-speech information and with a skeletal syntactic analysis.

But there are other pressing reasons to undertake such a project. Such data bases are of value for enterprises as diverse as the automatic construction of statistical models for the grammar of both the written and colloquial spoken language, the development of explicit formal theories of the differing grammars of writing and speech, the investigation of prosodic phenomena in speech, and the self evaluation of the adequacy of parsing models, the various formal syntactic theories embedded in those parsers, and the particular grammars of English encoded within those theories.

As a first step towards a much larger corpus, we have developed an annotation scheme for both part-of-speech information and higher-level syntactic structure, along with style books to assure consistent application of the annotation scheme, and have tagged a corpus of over 4 million words of contemporary English text with part-of-speech information, hand correcting the output of a stochastic part-of-speech tagger.

After early concerns about productivity, we investigated a range of methods for syntactic annotation (henceforth, tree banking) with respect to annotator speed, for annotators posteditting the output of Don Hindle's Fidditch parser. Key results:

1. Annotators take substantially longer to learn tree banking than the POS annotation task, with substantial increases in speed occuring after 2 months of training.
2. Annotators can postedit the full output of Hindle's parser at an average speed of 100-200 words per hour after three weeks, and 400-500 words per hour after two months.
3. Reducing the output to a far more skeletal representation similar to that used by the Lancaster UCREL TreeBank Project increases average speed to 700-750 words per hour. At this speed, a team of 5 part-time annotators working 3 hours a day should maintain an output of 2.5 million words a year of "treebanked" sentences, with each sentence posteditted by one annotator.

Treebanking has proceeded at full speed using skeletal annotation since December 1. We have annotated about 250K words of text, with 1/3 of this material bracketted by more than one annotator.

## References


[1] Eric Brill, David Magerman, Mitch Marcus and Beatrice Santorini. Deducing linguistic structure from the statistics of large corpora. In *Proceedings of DARPA Speech and Natural Language Workshop*, June, 1990.

[2] Fano, R. *Transmission of Information.* New York, New York: MIT Press, 1961.

[3] Harris, Z.S. *Structural Linguistics.* Chicago: University of Chicago Press, 1951.

[4] Harris, Z.S. *Mathematical Structures of Language.* New York: Wiley, 1968.

[5] David M. Magerman and Mitchell Marcus. Parsing a Natural Language Using Mutual Information Statistics. In *Proceedings of the Eighth National Conference on Artificial Intelligence.* July 1990, Boston, MA.

[6] David M. Magerman and Mitchell Marcus. Pearl: A Probabilistic Chart Parser. *Proceedings of the 1991 International Workshop on Parser Technology*, February 1991.

[7] Mort Webster and Mitchell Marcus. Automatic Acquisition of the Lexical Semantics of Verbs from Sentence Frames. *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics.* June, 1989, Vancouver, British Columbia.

# Learning Conventional Metaphors and Learning Using Conventional Metaphors

James H. Martin
Computer Science Department and
Institute of Cognitive Science
University of Colorado,
Boulder, CO
80309-0430

January 30, 1991

**Abstract**

Metaphor is a conventional and ordinary part of language. A computational approach to metaphor based on the explicit representation of knowledge about metaphors is presented here. This approach asserts that the interpretation of conventional metaphoric language should proceed through the direct application of specific knowledge about the metaphors in the language. MIDAS (Metaphor Interpretation, Denotation, and Acquisition System) is a computer program that has been developed based upon this approach. The focus here is on the learning capabilities of MIDAS.

## Research Interests

My current research interests lie in the area of conventional non-literal language. This area is concerned with the knowledge and mechanisms needed to adequately interpret language that deviates from what has traditionally been called literal. In particular, I am developing frameworks for dealing with the representation, use, and acquisition of knowledge used in the interpretation of idioms, metaphor, metonymy, and indirect requests. I am currently directing three projects in this area: the MIDAS system for interpreting metaphoric language, the METAMORPHOSIS learning system, and the METABANK, a empirically derived knowledge-base of English metaphorical conventions.

## References

[1] James H. Martin. *A Computational Model of Metaphor Interpretation.* Academic Press, Cambridge, MA, 1990.

[2] James H. Martin. *A Computational Theory of Metaphor.* PhD thesis, University of California, Berkeley, Computer Science Department, Berkeley, CA, 1988. Report No. UCB/CSD 88-465.

[3] James H. Martin. Representing regularities in the metaphoric lexicon. In *The Proceedings of the 12th International Conference on Computational Linguistics*, Budapest, Hungary, 1988.

[4] James H. Martin. Understanding new metaphors. In *The Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, Milan, Italy, 1987.

[5] James H. Martin. The acquisition of polysemy. In *The Proceedings of the Fourth International Conference on Machine Learning*, Irvine, CA, 1986.

# 1 Introduction

Metaphor is a frequent, systematic and conventional part of language. Natural language processing systems must be capable of dealing with metaphor in an effective way if further progress is to be made in applications like question answering, machine translation, and text summarization.

The main thrust of the approach to metaphor presented here is that the interpretation of conventional metaphoric language proceeds through the direct application of specific knowledge about the conventional metaphors in the language. Correspondingly, the interpretation of novel metaphors is accomplished through the systematic extension, elaboration, and combination of already well-understood metaphors.

Under this view, the proper way to approach the study of metaphor is to study the underlying details of individual metaphors and systems of metaphors in the language. This approach follows on the metaphor work of Lakoff and Johnson [7] and the computational approaches to metaphor described in [6, 8].

This approach has been embodied in MIDAS (Metaphor Interpretation, Denotation, and Acquisition System) MIDAS is a set of computer programs that can be used to perform the following tasks: explicitly represent knowledge about conventional metaphors, apply this knowledge to interpret metaphoric language, and learn new metaphors as they are encountered.

This knowledge-based approach to metaphor differs from the traditional computational account of metaphor. The traditional method is based on a problem solving paradigm. [1, 2, 3, 4, 5] The hearer's task is to use a problem solving strategy (typically analogical matching) to find or create the meaning of a metaphorical utterance from a representation of the literal meaning and more general world knowledge. While this approach does make use of a great deal of world knowledge, it does not make use of explicit knowledge about the metaphors that are conventionally a part of a given language.

The *metaphoric knowledge* approach, given here, raises a number of learning problems that do not arise in the traditional approach. Our current work is addressing the following two problems.

- How do language learners initially acquire the conventional metaphors that make up their language?
- How do these conventional metaphors effect the way that common-sense conceptual domains are acquired?

The next section will describe our efforts to address the first of these learning problems in the context of MIDAS. The following section describes our preliminary work on the METAMORPHOSIS system, which is intended to explore the second question.

# 2 MIDAS

This section provides a brief overview of the MIDAS approach to metaphor. In particular, it introduces the following issues.

**Representation:** The explicit representation of the conventional metaphors in a language in the form of explicit associations between concepts.

**Learning:** The dynamic acquisition of new knowledge about metaphors for which no known metaphor provides a coherent explanation.

## 2.1 Knowledge Representation

Consider the following simple example of a conventional UNIX metaphor. The metaphorical use of the word *in* reflects a systematic metaphorical structuring of UNIX processes as enclosures.

(1) I am *in* Emacs.

Metaphors like this may be said to consist of the following component concepts: a *source* component, a *target* component, and a set of conventional associations from the source to target. The target consists of the concepts to which the words are actually referring. The source refers to the concepts in terms of which the intended target concepts are being viewed. In this example, the target concepts are those representing the state of currently using a computer process. The source concepts are those that involve the state of being contained within some enclosure.

The approach taken here is to explicitly represent conventional metaphors as sets of associations between source and target concepts. The metaphor specifies how the source concepts reflected in the surface language correspond to various target concepts. In this case, the metaphor consists of component associations that specify that the state of being enclosed represents the idea of currently using the editor, where the user plays the role of the enclosed thing,

and the Emacs process plays the role of the enclosure. Note that these source-target associations are represented at the conceptual and not the lexical level. Any single lexical item or expression that can be construed as referring to the source concept of a known metaphor, may invoke that metaphor. In this example, the source component of the metaphor is attached to the concept of being enclosed, not to the lexical item *in*.

These sets of metaphoric associations, along with the concepts that comprise the source and target domains, are represented using the KODIAK [9] representation language. KODIAK is an extended semantic network language.

These sets of metaphoric associations representing conventional metaphors are full-fledged KODIAK concepts. As such, they can be related to other concepts and arranged in abstraction hierarchies using the inheritance mechanisms provided by KODIAK. The hierarchical organization of conventional metaphoric knowledge is the primary means used to capture the regularities exhibited by the system of metaphors in the language. Specifically, KODIAK is used to represent specialized domain specific metaphors, pervasive high-level metaphors, and the systems of relations among related metaphors.

## 2.2 Analogically Learning New Metaphors

MIDAS normally will locate and apply one these known metaphors to interpret text containing conventional metaphorical language. MIDAS will, however, inevitably face the situation where a metaphor is encountered for which none of its known metaphors provides an adequate explanation. This situation may result from the existence of a gap in the system's knowledge-base of conventional metaphors, or from an encounter with a novel metaphor. In either case, the system must be prepared to handle the situation. Consider the following example.

In this example, the user has employed the conventional UNIX metaphor that the termination of an ongoing process can be viewed as a killing. However, unlike the previous example, MIDAS finds that it is initially unable to interpret this example because it has no knowledge of this conventional metaphor. More precisely, it determines that the given input can not adequately satisfy the constraints associated with any of the concepts conventionally associated with the word *kill*.

```
> (do-sentence)
Interpreting sentence:
How can I kill a process?

Interpreting concreted input.

(A Killing16 (↑ Killing)
   (killer16 (↑ killer) (A I46 (↑ I)))
   (kill-victim16 (↑ kill-victim) (A Computer-Process10 (↑ Computer-Process))))

Failed interpretation:  Killing16 as Killing.
Failed interpretation:  Killing16 as Kill-Delete-Line.
Failed interpretation:  Killing16 as Kill-Sports-Defeat.
Failed interpretation:  Killing16 as Kill-Conversation.
No valid interpretations.
```

At this point, MIDAS has exhausted all the possible conventional interpretations of the primal representation. In particular, the direct non-metaphoric interpretation and three known metaphorical interpretations are rejected because their restrictions of the role of the `kill-victim` fail to match the semantics of the concept filling that role in the input, a `computer-process`.

This example illustrates the operation of the learning component of MIDAS, the Metaphor Extension System (MES). This system is invoked by MIDAS when it discovers a metaphor for which it has no adequate knowledge. The task of the MES is to attempt to extend its knowledge of some existing metaphor in a way that will yield a coherent interpretation for the new use and provide a basis for directly understanding similar uses in future. Analogical reasoning is at the core of MIDAS's learning mechanism. However, unlike previous metaphor systems, MIDAS does not attempt to draw an analogy between source and target domains of a metaphor. Rather, MIDAS attempts to reason analogically from known metaphors.

In this case, the system finds and extends a closely related known metaphor that also uses *kill* to mean a kind of terminate. MIDAS finds that there is a known metaphor covering the use of *kill* in *kill a conversation* to mean to terminate. This known metaphor is applied analogically to the current situation through the common notion of process meaning a series of related events happening over time.

```
============================================================
Entering Metaphor Extension System
============================================================

Attempting to extend existing metaphor.
Selecting metaphor Kill-Conversation to extend.
Attempting a similarity extension inference.
Creating new metaphor:  Killing-Terminate-Computer-Process

(A Killing-Terminate-Computer-Process (↑ Kill-Metaphor)
   (kill-victim-c-proc-termed-map kill-victim → c-proc-termed)
   (killer-c-proc-termer-map killer → c-proc-termer)
   (killing-terminate-computer-process-map Killing → Terminate-Computer-Process))

Final interpretation of input:

(A How-Q46 (↑ How-Q)
   (topic46 (↑ topic)
    (A Terminate-Computer-Process10
       (↑ Terminate-Computer-Process)
       (c-proc-termer10 (↑ c-proc-termer) (A I46 (↑ I)))
       (c-proc-termed10 (↑ c-proc-termed)
           (A Computer-Process10 (↑ Computer-Process))))))

UC: You can  kill a computer process by typing ^ C to the shell.
```

Finally, the target concept determined by the MES is used to provide an answer to the user.

The approach taken in MIDAS to the understanding of new or unknown metaphors is called the Metaphor Extension Approach. The basic thrust of this approach is that a new metaphor can best be understood by extending an existing well-understood metaphor or combining several known metaphors in a systematic fashion. Under this approach, the ability to understand and learn new metaphors depends critically on systematic knowledge about existing known metaphors.

This approach, therefore, shifts the processing emphasis in the case of novel metaphors away from the notion of attempting to determine the right target concept by a direct matching against the literal source. Rather, an attempt is made to determine the correct target through the use of an existing related metaphor. Therefore in this example, no attempt is made to find the intended target meaning by looking at the source details of literal slaying, rather the system examines the target concept of an already existing terminating as killing metaphor.

# 3 Metaphorically Learning New Concepts

Despite the demonstrated effectiveness of MIDAS's learning system, it clearly has a number of serious deficiencies. One major problem arises from the fact that while MIDAS relies heavily on the pre–existing conceptual representation of the various source and accomplish its learning task, it can not alter that representation in any way. Learning consists entirely of creating and storing new metaphors at various levels of abstraction.

To make this more concrete consider the following examples of the ubiquitous PROCESS-AS-ENCLOSURE metaphor.

(2) How can I get out of emacs?
(3) Get into vi to edit your .login file.
(4) I'm in mail.
(5) Tell me how to get out of lisp.

These examples illustrate the use of the widespread container metaphor in English. In this domain, this metaphor structures certain kinds of systems as environments. This concept of a system considered as an environment is not based on the particular functionality of the system but rather on the way that the user interacts with it.

Consider the following scenario, MIDAS is presented with a knowledge–base that classifies UNIX programs strictly according to their functionality. Assume further that the kb contains the specific metaphor EMACS-AS-ENCLOSURE, that structures EMACS as an enclosure. When (3) is encountered MIDAS can appropriately determine its meaning by analogy to the existing EMACS metaphor. It accomplishes this analogy by making use of the parent category

EDITOR shared by VI and EMACS. At this point MIDAS may appropriately create a more abstract metaphor EDITOR-AS-ENCLOSURE.

Continuing with this same scenario, consider what happens when MIDAS encounters (5). In the given functional hierarchy, LISP and the target concept of the relevant analog metaphor EDITOR-AS-ENCLOSURE are quite distant. They share the common ancestor concept UNIX-PROGRAM which dominates all the known programs in the knowledge-base. If this were used in a straightforward way as the basis for learning the meaning of (5) it would result in the creation of a UNIX-PROGRAM-AS-ENCLOSURE metaphor. The problem of course is that this is far too abstract and applies to many UNIX programs that simply do not permit this metaphor. To prevent this problem MIDAS only permits analogical generalization to occur when the common ancestor is extremely close to given analogs. (For example, VI and EMACS).

To summarize, this problem arises because MIDAS can not alter its representation of non-metaphorical domain knowledge. In this case, there is no appropriate abstract target concept to attach the new metaphor to. MIDAS must, therefore, either leave multiple metaphors at too specific levels of representation, thereby failing to capture a generalization, or it must place the metaphor at too high a level of representation potentially leading to an overgeneralization.

We are currently investigating these problems in the context of a system called METAMORPHOSIS. METAMORPHOSIS is a learning system that modifies the structure of a given knowledge-base under the influence of a conventional metaphor. As with MIDAS this investigation is situated in the domain of building natural language consulting systems for operating systems.

To make the task of METAMORPHOSIS more concrete, we will continue with our environment example. The system begins with a knowledge-base of facts about UNIX commands and programs. The knowledge-base is initially structured as an abstraction hierarchy with the various user programs classified according to their functionality. The task for METAMORPHOSIS is to create new categories that reflect the metaphorical structure of these concepts. It must perform this task by monitoring the language processing performed by MIDAS.

In our current example, the system's task is to create a new concept that roughly corresponds to the notion of an interactive system. This is the missing category in the target domain that dominates all and only those programs that permit the environment metaphor. This ultimately includes the editors, mail, and interactive language processors. The semantics of this category is determined by the meaning of the metaphors as determined by MIDAS. In effect, the conventional metaphors used by MIDAS are providing an inductive bias necessary for the creation of new categories. In these examples, the system notes that MIDAS is repeatedly applying a generic container metaphor to a subset of UNIX commands. Moreover, these metaphors are only being used to refer to certain aspects of these concepts.

# References


[1] Jaime Carbonell. Invariance hierarchies in metaphor interpretation. In *Proceedings of the Third Meeting of the Cognitive Science Society.*, pages 292–295. Cognitive Science Society, August 1981.

[2] Gerald F. DeJong and David L. Waltz. Understanding novel language. *Computers and Mathematics with Applications*, 9, 1983.

[3] Dan Fass. *Collative Semantics: A Semantics for Natural Language.* PhD thesis, New Mexico State University, Las Cruces, New Mexico, 1988. CRL Report No. MCCS-88-118.

[4] D. Gentner, B. Falkenhainer, and J. Skorstad. Viewing metaphor as analogy. In D.H. Helman, editor, *Analogical Reasoning.* Kluwer Academic Publishers, 1988.

[5] Bipin Indurkhya. Approximate semantic transference: A computational theory of metaphors and analogy. *Cognitive Science*, 11:445–480, 1987.

[6] Paul S. Jacobs. *A Knowledge-Based Approach to Language Production.* PhD thesis, University of California, Berkeley, Computer Science Department, Berkeley, CA, 1985. Report No. UCB/CSD 86/254.

[7] George Lakoff and Mark Johnson. *Metaphors We Live By.* University of Chicago Press, Chicago, Illinois, 1980.

[8] Peter Norvig. *A Unified Theory of Inference for Text Understanding.* PhD thesis, University of California, Berkeley, Computer Science Department, Berkeley, CA, 1987. Report No. UCB/CSD 87-339.

[9] Robert Wilensky. Some problems and proposals for knowledge representation. Technical Report UCB/CSD 86/294, University of California, Berkeley, Computer Science Division, May 1986.

**David M. W. Powers**
Universität Kaiserslautern
D-6750 KAISERSLAUTERN FRG

powers@informatik.uni-kl.de
Tel: (+49-631) 205-3449 Fax: (+49-631) 205-3200

Over the last 12 years, simulation research has been undertaken in Natural Language Learning in the context of an interdisciplinary theoretical framework made explicit in various hypotheses. The initial research was undertaken towards a PhD at the University of NSW, and was continued with a group of research students at Macquarie University (both in Sydney). At present Dr Powers is located at the University of Kaiserslautern, supported by ESPRIT BRA 3012: COMPULOG.

The first decade of this research, including a broad 'cognitive science' review, is presented in a monograph [Powe89] – with a further review in [Powe90]. Recent developments are presented here (overlapping with [Powe91]).

The main hypotheses explored are:

a) that the mechanisms responsible for language phenomena are more general than is often credited;
b) that the learning of language is inseparably tied to the learning of ontology;
c) that automatic self-organization and hierarchy formation can gives rise to a basic conceptual framework based on positive examples;
d) that the mechanism responsible for language learning phenomena have a considerable overlap with those for the rest of our sensory-motor experience;
e) that recognition and generation activities are logically and at least partially physically separate;
f) that the recognition components act as critics for the production components and provide negative information;
g) that interaction with the world also provides implicit negative information;
h) that contrast and similarity assessment of content in contexts provide a basic learning mechanism based on metaphor and paradigm;
i) that cognitive restrictions not only restrict our learning capability but the range of natural languages with the effect that our limitations actually assist the learning process;
j) that language should be examined from the perspective of ontological learning in an active environment;
k) that the concepts learned at one level are the symbolic building blocks for another level;
l) that the exchange of information with the environment as mediated by our sensory-motor system is not inherently different in form or representation from our higher level concepts.

Experiments fall into the following categories:

| | | |
|---|---|---|
| i) | neuro-visual association/learning | [Powe84,89] |
| ii) | mixed-mode parsing/learning | [Powe84,87,89] |
| iii) | word-class learning | [Powe84,87,89] |
| iv) | concurrent parsing/learning | [Powe84,87,89] |
| v) | statistical formula learning | [Powe84,89,91] |
| vi) | neural net formula learning | [Powe84,89,91] |
| vii) | critical formula learning | [Powe87,89] |
| iix) | critical semantic learning | [Powe87,89] |
| ix | ontological learning | [Powe89] |
| x) | morphological learning | [Powe91] |

In particular these batteries of experiments are all performed in multiple contexts, in the sense of one or more of: sensory vs linguistic modality, level of hierarchy, or various natural languages.

These hypotheses and results suggest that different forms of learning are appropriate at different levels of the language hierarchy which are characterized by different levels of input requirement, but that these forms of learning are widely applicable in terms of the precise domain of learning and that language learning must not be too narrowly characterized.

We summarize our tentative conclusions as follows:

1. at the lower levels, self-organization is achieved in the absence of a formal teacher and critic (in vision, orthography, phonology and grammar).
2. at intermediate levels, implicit teacher and critic can largely be provided by an active environment through multi-modal interaction;
3. at the higher levels explicit teacher and critic are helpful, but inessential.

## References

[Powe83] David M. W. Powers, "Neurolinguistics and Psycholinguistics as a Basis for Computer Acquisition of Natural Language," *SIGART* **84** 29-34 (June 1983).

[Powe84] David M. W. Powers, "Experiments in Computer Learning of Natural Language", *Proc. Aust Comp. Conf.*, pp489-499, 1984.

[Powe87] David M. W. Powers, L. Davila, D. M. Meagher, D. Menzies, "Further experiments in Computer Learning of Natural Language", *Proc. Aust. Joint AI Conf.*, pp458-468 , 1987.

[Powe89] David M. W. Powers, C. C. R. Turk, *Machine Learning of Natural Language*, Springer Verlag, London/Berlin, 1989

[Powe90] David M. W. Powers, "Goals, Issues and Directions in Machine Learning of Natural Language and Ontology". Chairman's background paper, AAAI Spring Symposium on *Machine Learning of Natural Language and Ontology*, Stanford CA (March 1991). Report SEKI-90-14, Univ. Kaiserslautern.

[Powe91] David M. W. Powers, "On the significance of Closed Classes and Boundary Conditions: Experiments in Lexical and Syntactic Learning", submitted to *IJCAI'91*.

# How far can self-organization go?

### Results in Unsupervised Language Learning

## Abstract

There are a number of debates in linguistic, psycholinguistic and neurolinguistic circles which have relevance to research on machine learning of natural language. Some of these concern where language lies on the spectrum between innate and learnt; how much can be learnt in the absence of semantics; how much can be achieved by neural self-organization without multi-layer back-propogation; and how important negative information is to language learning.

The computational research presented in this paper places a point of reference on each of these spectra, and indeed suggests that they are not independent.

We present some computational experiments and results, and propose ideas towards a theory of language learning. More importantly we pose some traditional questions in a new light and suggest new avenues of research for the traditional cognitive science disciplines as well as modern computational linguistics.

## Introduction

[Gold67] and [Mins69] produced results which demonstrated limitations on the possibility of learning. These were based on certain assumptions about the learning mechanisms and the problem domain, and were in various respects both intended and construed as criticisms of current approaches and claims. In the first case, [Gold67] showed that context free languages couldn't be learned without either a teacher or a critic. In the second case, [Mins69] showed that a class of (visually presented) group invariant relations could not be recognized by Perceptrons.

Since then, the more powerful PDP (Parallel Distributed Processes) approach popularized by [Rume86] (and subsequent publications from the same group) has demonstrated overwhelmingly that useful learning (*inter alii* in the language and vision domains) can be done with neural nets. In a less focussed way, MLNL (Machine Learning of Natural Language) has *also found renewed vigour [Lang87; Powe90].*

But there are still things our machines can't yet do. And there are still things our machines can't ever do. The results hold. But there are things we, that is humans and other organisms, can do. And there are language, vision and speech features that earlier statistical and neural models did learn [Koho84,89,90; Powe83,89; Ritt89]. The trick is to characterize these accurately and discover appropriate mechanisms – whether they be the *natural* mechanisms, just *similarly* effective mechanisms, or *better* mechanisms.

In [Powe83,89] one of several experimental language learning programs used self-organizing neural network techniques to learn word classes and syntactic rules in a total absence of critical input. There was simply multiple exposure to a set of legal phrases, with no teacher supplying anomalous input in the sense of [Gold67]. Nonetheless, the system managed to learn the word classes correctly, as well as grammatical rules which, if not actually those the grammarians discovered, are nonetheless effective. Similar results were achieved in a statistical program applied to the same data. The neural program was shorter. The statistical program faster.

[Koho90; Ritt89] independently showed that neural and comination statistical/neural self-organization techniques can learn word classes (but apparently *not* syntactic rules) of similar complexity in a different domain – again in the absence of critical input. (Similar techniques were applied by [Koho84] to mapping Finnish and Japanese phonemes – viz. achieving the feature/phone to phoneme classification.)

What is interesting is not just what was learned in terms of word classes, but what was learned first and why these particular rules were learned. It turns out that the most closed classes were learned first. These then seemed to act as pointers to the more open word classes they were associated with. This paper proposes that these results can give us insights as to why closed class words, such as articles, occur at all, how they are learned, and why they are not used early but are recognized. It also extends the experiments below the word level to see if there are closed classes there.

None of these previous reports or reviews has fully considered the broader computational, linguistic and psycholinguistic significance of these particular results (although [Powe90] does point to most of the issues involved). Here we consider this significance in several respects: in relation to closed classes, in relation to symbolic properties of connectionist systems, in relation to the weak form of learning used, and in relation to more accurate characterization of natural language.

Therefore, we will first summarize the methodology and results of the "noun phrase" experiments of [Powe84] and the "sentence" experiments of [Ritt89; Koho90], we then address some of the issues to which they are relevant and introduce some hypotheses to be tested. We finally present a computational experiment using similar techniques in the new, sub word-level, domain of classification of letters/phonemes into the classes from which syllables and words are composed, giving our procedures, results and conclusions.

## Previous Syntactic Learning Experiments

We do not wish to review statistical, neural or syntactic learning generally, but to take up certain experiments from [Powe89] and [Koho90], and compare the application of similar techniques in one of the domains that bridges the gap. As mentioned in the introduction, the pedigree of such work extends back beyond the criticisms of [Gold67] and [Mins69] and are reviewed and represented adequately elsewhere (see, in addition, [Lang87; Powe90] for pointers).

The experiments we wish to review were presented in the context *of noun phrases and filtered sentences; the classes categorized and* grammatical rules learnt were discovered with two different mechanisms, neither of which required critical input.

We imagine that the computational model represents a child at the beginning of the stage where he learns some nouns and verbs and their meanings and that he is trying to make sense at the same time of the images he is faced with. We further suppose that there are prosodic and syntactic features which tend to highlight the significant words, e.g. that they occur stressed in phrase final position. We hypothesize further that what is far beyond the child's competence and far from these significant positions is filtered out, and that conversely the child focuses on what is close to or within his competence.

We actually make no use of these assumptions other than to provide some justification for the type of dataset used for the learning experiments, which we present in figures 1 and 2 in the form used in the simulations of [Powe89] and [Koho90] resp.

```
the cat. #
a dog. #
my dog? #
this mat! #
...
```

**Fig. 1.** *Example dataset à la [Powe89].*

In the original experiments the '#' of Fig. 1 had some 'monitoring' significance and was not passed to the learning algorithm. It also serves as a reminder of the elision. The prosody of speech is hypothesized to have some correspondence to the punctuation symbols used in these text experiments.

```
Mary likes meat
Jim speaks well
Mary likes Jim
Jim eats often
...
```

**Fig. 2.** *Example dataset à la [Koho90].*

Note that both of these datasets can be regarded as sets of "three word sentences" representing utterances from which the uninteresting parts have been filtered according to different theories, or different applications of a general theory.

A first criticism can already be mentioned here: results with the omitted words included are *not* presented. Although the preliminary results from experiments with more complex data were (as could be expected) more complex and less conclusive, they would be interesting to see, and should give an idea of the

degree of reliance placed on the above-mentioned assumptions. (A listing of one of the actual neural programs used is however presented in [Powe89], allowing the possibility of repetition or extension of the experiment.)

It should be noted too that the learning, particularly for the (pure) neural simulations, is very slow. For example, the "semantic map" of [Koho90: Fig.12] resulted from "2000 presentations of word-context-pairs derived from 10 000 random sentences of the kind shown". (It is therefore very time-consuming and unrewarding to explore the more unlikely directions!)

### Statistical Psycholinguistic Model

The first model [Powe83,89] makes use of an additional psycholinguistic hypothesis. It uses the *Magical number seven plus or minus two* of [Mill56] to constrain the number of partial parse fragments (trees) kept around on *tags* and available for correlation. Unlike some of the earlier models, it then not only turns collocations of words into hypotheses of rules, but collocations of tags.

A second technique, also motivated by psycholinguistic considerations, is used to consolidate rules: in an induction step, bring together into the same hypothesized class words with collate similarly, viz. with the same words or classes. Thus classes are formed initially as small consistent cosets of words.

A thresholding step is used before rules are considered ready for *production* use – again a psycholinguistic hypothesis lies behind this terminolgy. It is proposed that the unthresholded grammar can play a role in guiding the recognition process in terms of indicating the likely class of a word, but that there is an implicit or explicit partitioning into *recognition* and *production* grammars mediated, in part, by some sort of threshold.

We present in Fig. 3 only a sample thresholded, consolidated grammar to give the flavour of the results.

The first observation to be made (to an extent observable in the structure of the rules) is that the first class learnt is the *punctuation/prosody*. Next come the *articles* and finally the *nouns*. The significant aspect is that the most *closed* (or smallest) classes are learnt first and that these act as pointers (in the rules) to the more significant *contentive* and *open* classes.

### Self-Organizing Neural Net

The above experiment was duplicated [Powe83,89] with a self-organizing model inspired by the visual application of such a neural net by [Mals73], but based in some respects on the model of [Klop82]. Interestingly, this program did not make use of the magical number seven directly, but a similar effect result from the *decay* model used. Once a neuron had fired it decayed over a period of time allowing for the possibility of it interacting with the neurons firing as a result of subsequent "words".

The results of this experiment were comparable with the statistical version, and a relationship between neurons and classes, synapses and grammatical rules was apparent in the comparison of the results.

The experiments of [Ritt89; Koho90] used a similar model applied to their dataset. For efficiency they turned to a hybrid statistical/neural approach in which they first preprocessed the data to produce an "average context" for each word – an average of all code vectors of predecessor-successor-pairs surrounding the given word. Note that this windowing is very sensitive to the omitted words, but could be justified on the basis that these words really represent the phrases of which those words are the nucleus.

The methodology of [Ritt89; Koho90] is explicitly exploiting the contextual similarity of items. The important feature is that the context is consistently dominant and recognizable in the learning process, and thus words may be classified by the contexts they occur in, and that then the classification of words together allows unification of contexts and consequent strengthening of the context consistency.

In these experiments the context taken was the pair of "words" preceding and succeeding the word in focus. In the experiments of [Powe84,89] the context was determined by the *decay* mechanisms or the *tag* mechanism. In recent experiments based on the paradigm of [Ritt89], similar results have been produced with "contextual sensitivity" being provided by the addition of recurrence between layers [Scho91].

## Hypotheses

The experimental perspective taken here is concerned with understanding the nature of language learning enough to implement useful models by whatever means, whether neural or statistical, hybrid or novel. And we follow [Powe89] in recognizing the importance of contributions from Cognitive Science, and our theoretical model conforms, in the main, to the hypotheses present in Chapter 13 thereof. In particular, we recognize the importance of physiological restrictions for the determination of the nature of language, we learn language by making hypotheses which can prove useful irrespective of their validity, we envisage the negative information necessary for learning as coming from the natural restrictions of human physiology, environment and current hypotheses rather than from explicit teachers and critics.

In neural networks this type of system behaviour is called *self-organization*. In other contexts it is called *auto-correlation* or *emergence*. It is can also be seen as a consequence of fundamental principles well known in Linguistics, and indeed the foundation of Phonology (and also its generalization to Tagmemics), namely: Contrast in Identical Environments (CIE) and Contrast in Analogous Environments (CAE).

We wish to develop one hypothesis further here. It is beyond the scope of this paper to go over once more the psycholinguistic evidence reviewed in [Powe89], but we note that the experiments we reviewed in the last section are consistent with, or at least suggestive of, the *complexity hypothesis*, *pivot grammars*, and *nucleus-margin coordination*. These suggest respectively that the simplest concepts (and by extension here, constructs and classes) are learnt first; that certain words in a *child grammar* function in a special way, as *pivots*, whilst not conforming precisely to adult *grammatical classes;* and that a *binary grammar* is evident, at many levels, in which the components differ in importance and may thus be designated as *nucleus* and *margin*.

In terms of gramatical *classes*, the natural complexity metric is the size of the class. A class that is always represented by a

```
Sense Class Thresh-Set

formula(lang, 24, [[17, 10]])
formula(lang, 17, [[12, 16]])
```

```
Sense Class Thresh-Set

class(lang, 16, [a, the,...])
class(lang, 10, [rat, cat,...])
class(lang, 12, ['.','?','!'])
```

**Fig. 3.** *Sample output from [Powe89a].*

single exemplar, or a very small number of exemplars, but whose degree of occurence is comparable with other classes, will clearly provide a unmistakable context which can act as a boundary condition for the self-organizing process. That is, *closed* classes will act as pointers to the more *open* classes. This facilitates *focussing* on the open class "word" and hence the attachment of semantics. The broader scope and easy identification of the *open* class therefore makes it the ideal candidate to be the main information carrier, or *contentive*, as well as the syntactic *nucleus*.

| Mem | (Lev) | Description |
|---|---|---|
| ?? | (0) | Several independent variables |
| 11 | (1) | 4 to 6 feature single phone characters |
| 10 | (2) | 2 or 3 character (C* or V*) clusters |
| 8 | (3) | 2 or 3 cluster C*V*C* syllables |
| 7 | (4) | 2 or 3 syllable morphs |
| 6 | (5) | 2 or 3 morph words |
| 4 | (6) | 2 or 3 word phrases |
| 3 | (7) | 2 or 3 phrase clauses |
| 2 | (8) | 2 or 3 clause sentences |
| 1 | (9) | 1 or 2 sentence (nuc./marg.) segments |
| .5 | (10) | 2 or 3 segment paragraphs |
| .2 | (11) | 1 or 2 paragraph monologues |
| .1 | (12) | 2 or 3 monologue dialogues |

**Fig. 4.** *Phono-morpho-phraseology.* Levels of the speech-language hierarchy, from feature level through Phonology and Morphology to Phrase Structure and Discourse Grammar are illustrated with a level number for reference and an idea of the possible variation of the number of units stored and available at that level (decreasing as complexity increases).

This process can be reflected at many levels, and is by no means limited to the speech hierarchy (Fig. 4). Similar processes were indeed first observed in vision [Mals73]. But in the context of speech, the prosodic features (including stress, intonation, speech rate and pauses) form clear easily distinguishable classes of limited membership. This allows focussing on phonological phrases and syllables. These have a close relationship to the grammatical phrase and morph, where a similar process can identify repeated syllable/morphs as contexts which will cohere into a closed class. Similarly phrases subtended by a particular closed class can act as units in which the frequently occuring templates can provide boundary conditions for the self-organization at that level.

The experiments reported above demonstrate these effects at several different levels. Phonemes have been mapped by neural self-organization; noun phrases have had their word components classified by the same and related statistical techniques; sentences have had their phrase/word components classified similarly.

We proposed to explore one of the missing pieces from this features to sentence classification: the syllable is normally defined in terms of particular patterns (varying according to language) or consonant (C) and vowel (V) classes. The syllable and these consonant vowel classifications are missing from the above demonstrations. The consonants and vowels are determined by phonetic features, and a related prosody also helps to identify syllables. Our theory would suggest that these physiological characteristics should act as restrictions (or boundary conditions) defining logical closed classes which would be actual syntactic entities, and would thus adopt also the associated syntactic and semantic properties (*open* = *contentive* = *nucleus*).

Why should we distinguish vowel and consonant – or indeed liquids, nasals, etc? Morphophonemics dictates some constraints, but why would we expect a grammatical function? This hypothesis provides an explanation. It further leads us to predict that we should discovering such a class by application of self-organization. To be more precise, we would expect the vowels to appear as a closed class rather than the consonants, being a smaller class – although liquids or nasals or something else could be a candidate according to size, but are excluded by their lack of primary grammatical significance. As there is not a one to one correspondence between phonemes and graphemes (characters) we allow the possibility of groups of graphemes to function as a unit, and hence the possibility that diphthongs or modified characters (e.g. +h, +r, +l, etc.) might be present.

There is also the question of how small a closed class should be – even those we have identified could conceivably be subclassified. We *need* not to introduce size as a parameter, however the *magic number seven* is again used as a memory/window constraint. The vowels happen, interestingly to fall into the *magic number seven plus or minus two* range. They may just be another addition to the catalogue of its magical properties!

## Algorithm

We first note that clusters or phrases (collationally significant class constructed from lower level units) are significant to the extent that:

*a*. Units occur relatively frequently with their predecessor(s);
*b*. Units occur relatively frequently with their successor(s);
*c*. Prefixed units have a modified class of successors;
*d*. Suffixed units have a modified class of predecessors;
*e*. Suffixed units have an almost unmodified class of successors;
*f*. Prefixed units have an almost unmodified class of predecessors.

Thus, /qu/ is significant by *a*, /th/ is significant by *b*, *c* and *e*, /ck/ is significant by *b*, *c* and *d*. In the case of properties *a* and *b*, one unit acts as a good predictor for the other member(s) of the cluster. Properties *c* and *d* indicate that the cluster does not simply inherit collations but has unique characteristics. The final pair of properties are related to apparent recursion, but are more general in that they extend to cohesive constraints.

Normally the modified succ/predecessors class is a reduction which excludes those which make up the other component of the structure. It may be that the class is as it would have been without the intervention (apart from such modifiers), or that it follows the modifier, or both. Thus /t/ can be followed by [h],[r],VOWEL; /th/ can be followed by [r],VOWEL; /tr/ can be followed by VOWEL; /thr/ can be followed by VOWEL. So: /th/ is a level 2 modification, /thr/ & /tr/ are level 3 clusters.

```
1. Read dict & produce Context-Char sets <= SEVEN
2. Significant sets -> Cluster-Cluster pairs
3. Group left & right sets as g & h distributions
4. Group complementary clusters into g & h cosets
  4a. Intersect gives distribution for both sides
5. Restrict all distribution size to SEVEN ± TWO
6. Autocorrelate for subset ± TWO of distribution
  6a. Intersect/Union for both/either side cosets
7. Make best SEVEN of Intersection/Union classes
  7a. Make mutually exclusive hyperclasses
```

**Fig. 5.** *Outline of algorithm.*

The present algorithm looks for signs of the first of these three pairs of properties: it collects all the contexts for each character and group of character within SEVEN character strings (including word boundary and capitalization codes); it then groups into classes all the common characters and character groups which occur in an identical context (left and right contexts separately), associating their sets of contextual distributions with the classes; it finally seeks to correlate similar distributions (± TWO) and allows evaluation according to either symmetric or assymetric

relevance, either weighted or unweighted by the size of the class found.

SEVEN and TWO are parameters which may be varied slightly. Examples of the results and the intermediate stage associations will be presented in the next section, along with some more detail concerning the transformations at each stage. An overview of the algorithm is presented in Fig. 5.

## Results

The first stage of the processing can be viewed as the construction of a finite state machine in which each occuring string of less than SEVEN characters constitutes a state and the following character occurences define a transition possibility. This representation was used for pragmatic reasons, including efficient indexing and other uses of the structure.

```
fsm(i,p,1,296).
fsm(v,a,1,297).
fsm(th,ə,2,299).
fsm(abl,ə,3,301).
fsm(g,i,1,308).
fsm(ab,l,2,309).
fsm('$co',n,3,310).
fsm(ra,n,2,310).
```

**Fig. 6.** *Finite State Machine representation of context and next character.* '$' marks a word boundary; '^' indicates the following character was upper case. Arguments are *context, focus, length of context, number of occurences in context.*

Examples are shown in Fig. 6 of the predicate *fsm*. Another predicate *gsm* provides a view of all pairs of clusters occuring with a combined total of SEVEN characters. Then for each left cluster the distribution of right clusters associated with it by *gsm* are extracted as *dgsm* and vice-versa (*dhsm*). A sample of these distributional classes is shown in Fig. 7, and it is already apparent there that the vowels, or something closely related, are a significant class.

```
dgsm(4,189,[d,l,n,r],'$^a').
dgsm(6,385,[a,ə,ər,o,r,u],'$^b').
dgsm(1,36,[r],'$^bə').
dgsm(5,326,[a,ar,h,l,o],'$^c').
dgsm(1,36,[r],'$^ca').
dgsm(5,198,[a,ə,i,o,u],'$^d').
dgsm(1,35,[l],'$^e').
dgsm(2,61,[r,rə],'$^f').
dgsm(1,20,[ə],'$^fr').
dgsm(4,144,[a,ə,o,r],'$^g').
dgsm(3,206,[a,ə,o],'$^h').
dgsm(3,106,[a,ə,o],'$^j').
```

**Fig. 7.** *Distribution classes subtended by a given left context (extract).* Extract is for word initial contexts from proper nouns. Arguments are *size of class, occurences of class+context, class, context.*

We now repeat the exercise with *dgsm* to group together the cosets of clusters which subtend the same distributional class, *cgsm*, and vice-versa (*chsm*). Although some small groups of very closely related clusters arise as cosets, as illustrated in Fig. 8, the sets can also often be described in terms of common initial or final segments (cp. properties *c* to *f* above). But as there are many similar distributional classes which are affected by sample error in the selection of a limited dataset as well as by memory constraints with the rejection of rare collations.

```
cgsm(1,458,[a,an,ə,i,ic,o,u],[pl]).
cgsm(1,2808,[a,ar,ara,as,at,ə,ən,ər,...],[$p]).
cgsm(2,1031,[a,ar,c,c^,ə,i,o,on],[$^m,^m]).
```

**Fig. 8.** *Cosets of left contexts subtending the same distribution class (extract).* Arguments are *size of coset, number of occurences, distribution class of clusters, coset of subtending clusters.* Quotation marks are omitted for compactness.

So far we have performed *Contrast in Identical Environments* (CIE) type classification, now we want to perform *Contrast in Analogous Environment* (CAE) type classification to bring together similar distribution classes and combine their cosets and assess the number of different collations and occurences for these fuzzier hypersets of distribution classes.

```
classg(h,[$a,a,ə,i,mi,o,u],[a,ə,i,o,u],[$^d,...]).
classg(h,[$a,a,ə,i,mi,o,u],[a,ə,i,o,u],[cr]).
classg(h,[$co,$i,a,co,i,o,u],[a,co,i,o,u],[s]).
classg(h,[$co,$i,a,co,i,o,u],[a,co,i,o,u],[n]).
```

**Fig. 9.** *SEVEN classes (right) and close intersections with left distribution classes (extract).* Distribution classes of size SEVEN±TWO are used to find other distribution classes which are similar in that the intersection with the SEVEN class differs by no more than TWO from the SEVEN class. Arguments are *source of selecting SEVEN class, SEVEN class, intersection with distribution class, coset of distribution class.* Quotation marks are omitted for compactness.

In fact, we use the sets of known distribution classes intersected with themselves to define a kernel which must be within TWO of the size of the intersecting class. For efficiency, we use as intersecting classes only those with a size in the SEVEN±TWO range. As illustrated in Fig. 9, the vowel class emerges as one of the most important of these.

At this point, we combine the information from left and right distributions and compute statistics based on the size of the common and total cosets of the SEVEN classes, or the number of actual occurences of subtended collations. On all four metrics, the vowels emerge as the most well defined class – with a significant lead over the runner up in second place, as shown with best seven scores for two of the metrics in Fig. 10.

## Conclusions

In these experiments using statistical techniques and a single exposure to each word of the Unix dictionary, the vowel class emerged first, suggesting it as a closed class. The cosets were primarily consonant clusters, suggested analogously as an open class. This confirmed a prediction that the vowel-consonant distinction was of significance in learning, that the vowels would emerge as a closed class providing a limited number of contexts, and that consonant clusters would emerge as open classes.

One surprise was that diphthongs were not represented, and indeed vowel-semivowel collations came nearer to achieving membership.

We suggest that the magic number seven plus or minus two [Mill56] should also encompass the number of the vowels. It was indeed a parameter in the analysis, and variation of this parameter did vary the precise class learnt, but the relationship has not yet been analyzed. However, its application to the size of the selected class seemed least decisive – similar results were achieved with 6±2 and 7±3 settings, for example.

The exclusion of diphthongs may also be an indicator that they are recognized as complex, at least in the orthography and under

```
coseti(28,84,4,12,[a,e,ea,i,in,o,u],[d,n,s,t],['$1',b,c,d,h,l,n,p,r,s,st,t]).
coseti(28,112,4,16,[c,f,g,p,s,t,v],[a,e,i,o],['$a','$re',^,a,al,an,e,en,er,i,...]).
coseti(30,144,5,24,[c,d,g,l,s,t],[a,ar,e,l,o],['$a',^,a,an,ar,e,en,er,i,in,l,...]).
coseti(30,168,5,28,[a,e,i,o,u,y],[b,c,m,p,s],['$h','$m','$s','$t',^,an,b,c,...]).
coseti(48,156,8,26,[a,e,er,o,r,u],[b,c,e,f,g,i,n,t],['$^b','$f','$p','$t',...]).
coseti(49,196,7,28,[a,e,i,o,r,ra,u],[b,c,d,f,g,r,t],['$b','$c','$d','$g',...]).
coseti(85,385,17,77,[a,e,i,o,u],[b,c,ch,d,e,f,g,l,ll,...],['$^d','$b','$c','...]).
```

**Fig. 10a.** *Cosets of SEVEN classes of either context sorted by occurence in intersection (extract).* Arguments are *occurences of intersection coset, occurences of union coset, size of intersection coset, size of union coset, SEVEN class, intersection coset, union coset.*

```
coseti(30,168,5,28,[a,e,i,o,u,y],[b,c,m,p,s],['$h','$m','$s','$t',^,an,b,c,...]).
coseti(49,196,7,28,[a,e,i,o,r,ra,u],[b,c,d,f,g,r,t],['$b','$c','$d','$g',...]).
coseti(3,96,1,32,[a,e,o],[y],['$^g','$^h','$^j','$^p','$^s','$cr','^g','^h',...]).
coseti(16,184,4,46,[a,e,o,u],[i,ll,mp,ri],['$^b','$^d','$ch','$l','$m','$n',...]).
coseti(15,245,3,49,[a,e,i,o,r],[ch,t,th],['$b','$c','$d','$f','$g','$p',...]).
coseti(16,232,4,58,[a,e,i,o],[k,sp,u,v],['$^l','$^m','$^n','$^r','$br',...]).
coseti(85,385,17,77,[a,e,i,o,u],[b,c,ch,d,e,f,g,l,ll,...],['$^d','$b','$c',...]).
```

**Fig. 10b.** *Cosets of SEVEN classes of either context sorted by size of union (extract).* Arguments are *occurences of intersection coset, occurences of union coset, size of intersection coset, size of union coset, SEVEN class, intersection coset, union coset.*

the assumptions behind this program. Recent psychological studies indicate that familiarity with written language may necessary to the (conscious) recognition of segments [Read86; Mann86]. But are diphthongs recognized as complex? Are vowels recognized as having features? Is this totally acoustic or does it have a motor component? It will be very interesting to see what results of similar experiments achieve on speech!

Although this experiment was performed using statistical techniques rather than neural networks, it was guided by previous work which achieved similar results using either or a mix, and it is expected that similar results could straightforwardly be achieved in a neural simulation.

The success of back-propogation in multi-layer neural nets has perhaps overshadowed self-organization in simpler networks, despite the impressive early low-level results; the need for semantics has perhaps overshadowed the internal consistency of grammar at the lower levels; the theoretical need for negative information from the environment has perhaps overshadowed the effective supply of criticism from boundary conditions and system restrictions; and more generally the tendency to assume that basic linguistic distinctions are innate and very closely tied to the perceptual system itself may overshadow the fact that some of these distinctions can be learnt very easily with very basic mechanisms. These alternative perspectives are worthy of more emphasis and study.

This paper has presented some computational results and hypotheses about language learning. More importantly it poses some traditional questions in a new light and suggests new avenues of research for the traditional cognitive science disciplines.

## References


[Gold67] E. M. Gold, "Language Identification in the Limit", *Information and Control* **10** 447-474 (1967).

[Koho84] T. Kohonen, K. Mäkisara, and T. Saramäki, "Phonological Maps - insightful representation of phonological features for speech recognition", *Proc. 7th Int. Conf. on Pattern Recognition* 182-185 (Montreal Canada, 1984)

[Koho89] Teuvo Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, BERLIN FRG (3rd edn, 1989)

[Koho90] Teuvo Kohonen, "The Self-Organizing Map", *Proc. of the IEEE* **78** 1464-1480

[Klop82] Klopf, A. Harry, *The Hedonistic Neuron: A Theory of Memory, Learning and Intelligence*, Hemisphere, WASHINGTON DC (1982).

[Lang87] P. Langley, "Machine Learning and Grammar Induction", Editorial to Special Issue, *Machine Learning* **2** 5-8 (1987)

[Mals73] Malsburg, C. von der, "Self-Organization of Orientation Selective Cells in the Striate Cortex", *Kybernetik* **14** 85-100 (1973).

[Mann86] Virginia A. Mann, "Phonological awareness: The role of reading experience", *Cognition* **24** 65-92 (1986).

[Mill56] George A. Miller, "The magical number seven, plus or minus two: some limits on our capacity for processing information", *Psychologcal Review* **63** 81-97 (1956)

[Mins69] Marvin Minsky and S. Papert, "Perceptrons", MIT Press (1969).

[Powe84] David M. W. Powers, "Experiments in Computer Learning of Natural Language", *Proc. Aust Comp. Conf.*, pp489-499, 1984.

[Powe89] David M. W. Powers, C. C. R. Turk, *Machine Learning of Natural Language*, Springer Verlag, London/Berlin, 1989

[Powe90] David M. W. Powers, "Goals, Issues and Directions in Machine Learning of Natural Language and Ontology". Chairman's background paper, AAAI Spring Symposium on *Machine Learning of Natural Language and Ontology*, Stanford CA (March 1991). Report SEKI-90-14, Univ. Kaiserslautern.

[Read86] Charles Read, Zhang Yun-Fei, Nie Hong-Yin and Ding Bao-Qing, "The ability to manipulate speech sounds depends on knowing alphabetic writing", *Cognition* **24** 31-44 (1986).

[Ritt89] H. Ritter and T. Kohonen, "Self-Organizing Semantic Maps", *Biol. Cyb.* **61** 241-254 (1989)

[Rume86] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*, MIT Press, Cambridge MA (1986).

[Scho91] J. C. Scholtes, "Learning Simple Semantics by Self-Organization", submitted to AAAI Spring Symposia on *Machine Learning of Natural Language and Ontology* and *Connectionist Natural Language Processing*, Stanford CA (March 1991).

# Larry H. Reeker

Institute for Defense Analyses
Computer & Software Engineering Division
1801 N. Beauregard St.
Alexandria, VA 22311-1772
reeker@ida.org

Telephone (703) 845-6621 Fax (703) 845-6848


My relevant background includes a PhD in Computer Science at Carnegie-Mellon, following a BA in mathematics and philosophy - with a strong dose of linguistics - at Yale. I studied a lot of additional linguistics over the years during graduate school, at Linguistic Institutes, and while I was teaching. I have held faculty positions (jointly) in linguistics departments, but primarily have been a computer scientist, working in theory of computing, programming languages and systems, and artificial intelligence. I also headed computer science departments at Queensland (Australia) and Tulane and have worked as a technical executive in a private corporation and on the research staff of IDA, a not-for-profit "think tank".

Among the research topics on which I have worked over the years, my favorite is probably language learning, which is why I have pursued approaches to modeling it off and on for the last twenty years. I won't go into the details of my early syntactic acquisition models, which are mentioned in the *Handbook of Artificial Intelligence* and elsewhere, but list some relevant publications below and will be glad to discuss them or send information to anyone interested. The models were implemented computationally in a set of SNOBOL4 programs, all called PST, but actually a succession of elaborations on a single learning paradigm. In each case, the program was presented sentences, along with a meaning representation for each sentence. If it could not process the sentence with its available grammar, it tried to understand some portion of the sentence. If a certain degree of understanding was achieved, even at the single lexical item level, some modifications of the programs internal grammar could take place. The programs only addressed this issue of possible adaptation, not the issue of how lexical semantics are learned or how many trials it might take to learn (and possibly unlearn) new structures.

Some years ago, I commented to Herb Simon on my disappointment that there was no way to verify that the models, which acted plausibly in computer simulations, really had anything to do with human language learning, since the detailed data about understanding in human infants was not available, and observation may give a false picture of the extent of understanding (see Carol Chomsky's results and others). He suggested that one might consider developing a second language tutoring system that could, through built-in tests, give a view of how the language was being acquired, based on known exposure to the second language, especially if the language was a sufficiently "exotic" one that the learner's exposure was entirely or predominantly through the system. Leaving aside the possible differences between first and subsequent language acquisition, this would certainly give some clues to human language acquisition. Although I proposed to develop such a system and wrote some programs (again in SNOBOL4) for individual modules, I never had the time or funding to do so, except for early design (see the Australian paper cited below), but have not given up the thought.

A few years ago, I floated an idea that again put the machine in the position of learner, the idea being to use the human as tutor and a computer program as the learner. The idea got some interest from a government laboratory (the U.S. Air Force Human Resources Laboratory) and a limited amount of funding, but I was unable to follow up on it because of more pressing items. More recently, I was given funding by the Applied Information Technologies Research Center of Columbus, Ohio, to work the concept into a system design. That is the topic of my paper at the Spring Symposium, and I believe that it offers both a desirable and practical device for human-computer interaction and an approach to understanding the language acquisition process, based on dialogue between the learner and speakers.

**Related Publications (In Addition to Those Cited in the Following Working Paper)**

**Early Syntactic Acquisition:**


A Problem Solving Theory of Syntactic Acquisition, PhD Dissertation, Department of Computer Science, Carnegie-Mellon University, 1974.

The interplay of semantic and surface structure acquisition, *Recent Advances in the Psychology of Language* (Campbell and Smith, eds.), Plenum Press, vol. 2, pp.71-90, 1978.

Varieties of learning in grammatical acquisition, *Structural/ Process Models of Complex Human Behavior* (Scandura and Brainerd, eds.), Sijthoff and Noordhoff, pp.465-477, 1978.


**Second Language Acquisition:**


An artificial intelligence approach to natural language teaching, *Proc. 3rd Australian Computer Science Conference*, Australian National University, Canberra, pp.97-106, 1980.


**Language Learning Computer Interface:**

(See following paper and references therein).

**Miscellaneous Related Topics (Selected Papers):**


Artificial intelligence -- a case for agnosticism (with B. Chandrasekaran), *IEEE Transactions on Systems, Man and Cybernetics*, vol. 4, no. 1, pp. 88-94, 1974.

An extended state view of parsing algorithms, *Papers in Computational Linguistics*, Hungarian Academy of Sciences, pp. 141-160, 1976.

Natural language devices for programming language readability: embedding and identifier load, *Proc. 2nd Australian Compute Science Conference*, Hobart, pp. 159-167, 1979.

Some results on pure grammars, *Proc. 2nd Australian Compute Science Conference*, Hobart, 53-71, 1979.

An experimental applicative programming language for linguistics and string processing (with P. A. Bailes), *Proc. 8th Int'l Conference on Computational Linguistics*, Tokyo, 520-525, 1980.

Specialized information extraction: Automatic chemical reaction coding from English descriptions (with E. Zamora and P. Blower), *Proc. Conf. on Applied Natural Language Processing*, pp. 109-116, ACL, Santa Monica, 1983.

Specialized information extraction from natural language texts: The "Safety Factor", *Proc. 1985 Conf. Intelligent Systems and Machines*, 318-323, Oakland University, 1985.

Pattern-directed processing in Ada (with Kenneth Wauchope), *Proc. 2nd IEEE Int'l Conf. Ada Applications and Environments*, Miami, 49-56, 1986.

# Language Learning and Adaptive User Interfaces

Larry H. Reeker
Institute for Defense Analyses
Alexandria, VA USA

## Abstract

An adaptive user interface is one that changes its behavior to accommodate the preferred interactive behavior of the user. This paper discusses the concept of interfaces that adapt to the linguistic idiosyncracies of the user. It discusses two slightly different approaches, both of which have advantages, and argues for the utility of adaptive interfaces.

### 1. Introduction: Interface Adaptivity

If the objective of a user interface is to allow communication with a system in the form most natural to each of a broad range of users, a form that may include natural or artificial languages or a mixture of the two, then the interface must be adaptive. In other words, it must be able to learn to follow the user's commands that it cannot initially understand. It must also be individualized, since all users will not want it to adapt in the same way.

### 2. The Advantages of Adaptivity?

Adaptation has advantages over a natural language interface. The problems of natural language communication with computers are well-known (see, for example, [Reeker, 1980]) and have not really been alleviated by advances in natural language processing technology. Watt [1968] perceptively pointed out over twenty years ago the fact that a natural language interface that is imperfect may decrease the user's ability to stay within the bounds of the acceptable interface language (which we will call the **system language,** contrasted to the **user language**). Natural language output may do the same, encouraging the user to expect a greater breadth of linguistic understanding than actually exists in the system.

There are a number of advantages of a system's adaptation to the individual user. For occasional users, it simplifies the process of having to relearn the system each time it is used. Even regular users feel more comfortable with a "personalized computer" that reacts appropriately to their idiosyncratic usages, as evidenced by the fact that seasoned computer programmers develop ways to adapt their systems to their preferred usages. Some systems provide means to facilitate manual adaptation in limited ways, but adaptive systems do it automatically. Thus experienced users can consciously change the particulars of interaction as they become more practiced, and the system will adapt to the new modes of interaction. Adaptive interfaces have been proposed as an alternative that has advantages over a monolithic interface, natural language or otherwise [Reeker, 1984; Lehman, 1989].

The argument for adaptivity is made strongly by Jill Fain Lehman [1989; 1990b], who examined some of the implicit behavioral assumptions underlying the argument for the benefits of adaptive interfaces. The assumptions are:

(1) The user's linguistic interaction with an adaptive natural language interface will be consistent enough to arrive at a relatively stable common language for user/system interaction.

(2) This language will differ from user to user in significant ways.

(3) The user's ability to use more individualized, idiosyncratic language will result in better task performance than having to use a built-in interface language.

To test the hypotheses that these assumptions were valid, Lehman set up experiments using a simulated adaptive interface with a hidden operator. The simulation was based on the design for a real system, subsequently implemented as CHAMP (discussed below). Her results indicated that the users did exhibit individual consistency and comparative variability. As might be expected, not all users had the same difficulty in adapting to the built-in interface language, but some clearly did, and initial problems did seem to be overcome by (simulated) system adaptation.

### 3. A Design for an Adaptive Individualized User Interface

In proposing adaptivity, it may seem that we are suggesting a task more difficult than building a natural language interface, since building systems to learn a cognitive skill is generally more difficult than constructing a system to perform that skill. But the task of developing an adaptive interface, while not simple, is facilitated by the use of a good deal of built-in knowledge. It is not as difficult as building a natural language interface that will be adequate for all users (if one can exist), and clearly less difficult than building a system for learning a natural language *ab initio*, though it is related.

A slightly different Adaptive Individualized User Interface (AIUI) was designed by this author, using a transformational approach. In the AIUI, the system is allowed a large amount of built-in knowledge, including a rich lexicon and syntax of the user inputs expected and a semantic mapping for the built-in inputs that the system can deal with (system inputs). Based on discovery of the meaning equivalence between novel user inputs and known system inputs, the AIUI has to formulate translations which will be learned in generalized form as a set of linguistic transformations.

The known, constrained domain of discourse that takes place at a particular system interface provides an opportunity to use what Rada (Forsyth and Rada, 1986) calls "knowledge-rich" learning strategies, provided some additional knowledge can be obtained from the user when necessary. In the AIUI, this knowledge is obtained by a user-machine dialogue. And the dialogue quickly zeros in on the known knowledge because the machine knows what it needs, the user is cooperative, and the machine has been furnished with enough knowledge to conduct the dialogue (some of it heuristics based on the domain). The dialogue does not have to take place every time a user uses a new utterance, since there are a finite number of different forms (though, for practical purposes, there are not a finite number of different utterances), and the machine will generalize over forms.

The AIUI was designed for system inputs consisting of a limited set of UNIX commands (not a finite language, since any file names can be used), and the user inputs may be those commands or English versions thereof, or combinations of English and UNIX. The choice of this domain was made because of the constrained domain of discourse, rather than the prospective utility of an interface to UNIX, since the system was experimental.

Space does not permit details of the overall workings of the AIUI in this paper, but a design summary and discussion of the underlying linguistic framework will be presented. Further details can be found in the reports on the design project [Reeker, 1988].

The AIUI contains, for a given user, a User Transformation Dictionary (UTD). When user commands are not legitimate system commands as given, the UTD is consulted to look for candidate transformations to system commands. If no such transformation exists, the AIUI will try to adapt. The core functional module of the adaptive process must therefore find a translation of the user input and execute the command. If the translation and its effect meet the approval of the user, then a generalization process takes place and the resulting transformation is stored in the UTD.

A summary of the processes by which the AIUI deals with user commands that are not in its system command repertoire is given in Table 1. (Due to implementation considerations, this is not the actual flow in the program, but is easier to conceptualize.)

Transformations and parsing are discussed below.

TABLE 1. SUMMARY OF PROCESSES IN AIUI
(When the User Command is Not a System Command)

Case I: User Command Can be Parsed
- Case I-A: Transformation in UTD
  - Apply Transformation
    - Match
    - Transform
- Case I-B: Transformation Not in UTD
  - Case I-B-1: Meaning of User Assertion Known to AIUI
    - Formulate Transformation
      - Specific Tree Mapping
      - Category Generalization
      - "Risky Generalization"
    - Organize Transformations
  - Case I-B-2: Meaning of User Assertion Not Known
    - Heuristics
    - Partial transformation

Case II: User Command Cannot be Parsed
- Case II-A: All Lexical Items Are Known
  - Partial parses
  - Formulating New Transformations
  - Reformulating Grammars
- Case II-B: There Are Unknown Lexical Items
  - Assign new categories
  - Merge categories that use same Transformations
- Case II-C: Create Transformation for Given String Only

## 4. The AIUI in a Linguistic Framework

### 4.1. General Description

There is still controversy about how to process language, in terms of the stress placed on structural processing or direct meaning processing (proceeding from the lexical semantics). The view taken in the design of the AIUI is that either approach can give enough information to at least begin a dialogue between a person and a machine that will lead to the desired communication, and that both are needed if the system is going to adapt to the user. The knowledge obtained by adaptation is stored in terms of structural information and semantic mapping information, as described below.

### 4.2 The Transformations Used

The process of translation from user input (user command) to one the system understands (the proper system command to convey the meaning of the user command) is driven structurally but incorporates the semantic mapping through a set of transformations. The transformations have something in common with Chomsky's early theories of transformational grammar [1957], in that there is a kernel language consisting of the built-in system language, with the rest derived by transformations. Given the fact that the kernel sentences can also be considered as the meaning representation language for all intelligible inputs, the linguistic model begins to look much more like that found in theories of generative semantics, in which transformations were from the meaning representation to the surface structure, without the intervention of a (syntactic) deep structure (see e.g. the exposition in [Grinder and Elgin, 1973]). It was never clear just what the meaning representation should look like, and today, meaning representation is recognized as the major problem in natural language processing. But the meaning representation for the AIUI is clear, so the theoretical framework follows naturally.

Since the purpose of transformational grammars is generally to define the well-formed sentences of the language, they are formulated as transforming structural descriptions of kernel sentences to structural descriptions of surface sentences; but in the AIUI, transformations are used for translation to kernel sentences, so they are formulated in the opposite direction. The grammar used to parse input sentences is context free, with categories that reflect input expectations, based on the domain semantics. Because the system commands are all legitimate user commands, they are part of this grammar. A sample initial grammar and lexicon are shown in Tables 2 and 3.The types of user inputs (and their meanings in terms of system commands) that this grammar could treat is shown in Table 4.

### 4.3 Some Comparisons to CHAMP

Lehman's interface, CHAMP, does not use transformations, but learns phrase structures in a way that is quite analogous to the approach to early syntactic acquisition by

Table 2. Illustrative Partial Grammar for Example

| | |
|---|---|
| Σ → <C> | <C> → <CT>and <CP2>\|<CT> |
| <C> → <CP> \| <CP><CA> | <CT>→<OV><CTP><CPP> |
| <CP>→<CP><PIPE><CVP>\| <CVP> | <CVP><CTP>→<CV'>and <CTP>\|<CV'> |
| <CV> → <CV> <AP> \| <V> | <CV'> → <V><VPP>\| <V> |
| <CA →<CVA> <FILE> | <VPP> → <PWITH><ARGS> |
| <AP>→<ARG><AP>\| <ARG> | <ARGS>→<ARG>and <ARGS>\|<ARG> |
| <CVP> →<CVP><FILE> \| <CV> | <CPP> → <PON><FILE> |
| BOLDFACE indicates a rule that generates a basic UNIX command form. (i.e. System Grammar rule). | <CP2> → <CV2> <CPP2> |
| | <CPP2> →<PTO><FILE> |

Table 3 . Illustrative Partial Lexicon for Grammar of Table 2

| | |
|---|---|
| <CVA>→ > \| >> \|> | <PTO> → to \| on \| ... |
| <PIPE> → \| | <ARG> → -l \| -r \| ... |
| <OV> → run \| do \| ... | <CV> → s \| troff ... |
| <PWITH> → with \| using \| ... | <ARG>→ -l \| -r \| ... |
| <CV2> → append \| send to \| ... | <V> → ls \| troff ... |
| <FILE> → aitrc.1 \| aitrc.2 ... | |

Some of the categories in thesystem are defined by morphology or by situation (e.g. whether a file or directory is in the currently accessible directory structure), rather by listing in the lexicon.

this author [Reeker, 1970, 1974, 1975]. In that approach, a memory-limited bottom-up parser worked on adult input sentences, producing reduced forms (simulating the baby's impoverished short-term memory). The results were compared to sentences in the child's grammar and changes were made in single rules and in their corresponding semantic mapping rules, with generalization constrained by coherence and consistency criteria.There is a difference, however, both in the assumption as to inputs and the richness of the grammatical apparatus. CHAMP's parser is semantically and pragmatically constrained. In the child language model, the semantics, which were attribute type with complex (tree-form) attributes and composition rules, were separate from the grammar and constrained only the acquisition, based on the second input, which was the meaning. The view of intermixed structural determination and meaning determination was not yet current at the time of the earlier work, but seems more realistic

So in CHAMP the system uses the user's current grammar and lexicon to try to parse the user input, and Lehman's adaptive parser design classifies common deviations (ones recoverable by insertion, deletion, substitution, and transposition) in terms of degree of deviance. If the deviance is zero (the user input is parsable already), there is no adaptation required. If it finds one or more parses that are deviant by some amount but not more than a threshold value (which was two in the experiments mentioned earlier), then it classifies the user input as learnable. Learning consists of adding lexical items and/or rules to the user's grammar and generalizing properly. These are all discussed in Lehman's thesis [1989] and machine learning paper [1990a]. There is a similarity to conditions discussed for learning AIUI [Reeker, 1988; Reeker and Morrison, 1988], though the details are different because of the different theoretical framework.

Table 4. User Commands and Meanings (System Commands)

| User Command | Meaning |
|---|---|
| (1) run vi on aitrc.1 | vi aitrc.1 |
| (2) run vi with l on aitrc.1 | vi –l aitrc.1 |
| (3) run tbl and troff on aitrc.1 | tbl aitrc.1 \| troff |
| (4) run vi with r and x on aitrc.1 | vi –r –x aitrc.1 |
| (5) run ls with l and grep with e"d" on aitrc.1 | ls –l aitrc.1 \| grep –e"d" |
| (6) run troff on aitrc.1 and append to aitrc.2 | troff aitrc.1 >> aitrc.2 |
| (7) run ls with l and grep with e"d" on aitrc.1 and append to aitrc.2 | ls –l aitrc.1 \| grep –e"d" >> aitrc.2 |

Informationally, it can be shown that the additions of rules in a non-transformational generative grammar of sufficient power and the corresponding addition of transformations is equivalent. Given the form of the grammar in CHAMP, even the semantics carried in the transformations of AIUI has an equivalent form in the grammar, and things like context sensitive syntax can be dealt with by constraints. This has an appeal, since the use of a body of transformations carries a time penalty. There are no pragmatic constraints in the linguistic apparatus of AIUI, which may give an advantage to CHAMP. Pragmatics are used in the heuristics of AIUI, but the user could use a command with a meaning that is totally strange in the pragmatic context, and the only problem would be that the system would have difficulty in discovering the meaning. The role of pragmatics is an interesting one, as is the question of which framework is preferable for the grammar. Some experience with the systems will help to answer questions about these things.

### 4.4.Building-In Structural Knowledge

As suggested earlier, it is desirable to have as much understanding of the structure of anticipated inputs incorporated into the system as possible. Thus a rich initial grammar of user commands is built up by starting with the syntax of the kernel of system commands and adding expected structures from English. The grammar given in the design report may not be ideal in that respect, but there was an attempt to do it systematically. The grammar for system commands was developed on a semantic or logical basis, based on binary divisions of the domain of expressive possibilities. The grammar for user commands was developed to mirror English usage and to map into this logical format. Being keyed to the sublanguage and to the mappings to system commands, it carries a lot of semantics in its rules. It is anticipated that the development of grammars for adaptive interfaces, like other aspects of applied system design, would become easier as experience was gained with the systems.

### 4.5. Determining Meaning Through User Dialogue

The AIUI requires that the system be supplied with the meaning of an input that it does not understand. Heuristics for determining the meaning of user commands depend on the limited universe of discourse and pragmatics of the interaction. As an example, consider a user command "run vi on aitrc.1", the system might not be able to parse it in full initially, but would still be able to assign to vi the syntactic

category <CV>. It is likely that the syntactic category of aitrc.1 can be guessed also. First, vi will require a following file name. It is likely that the file aitrc.1 will appear in the current directory. If a path name had been given (like "dir1/dir2/foo") then it is even more clearly (by morphological criteria) a file. So the user command has within it a UNIX command verb and a file. The first hypothesis has to be that the user command means "vi aitrc.1". When the user verifies this, the system can hypothesize a transformation. A series of such heuristics has been derived for the AIUI. Again, the heuristics are based on the domain of discourse and must be derived individually; but the richer they can be, the better the interface will be.

If the user were to type a "hybrid" command like "troff aitrc.1 and place in y", a similar dialogue would determine that this means "troff aitrc.1 > y". The system merely assists the user in recognizing that this is the form meant, then enters a transformation. The user cannot, on the one hand, expect to use the system without any knowledge of the commands. Of course, on the other hand, if the user knows the UNIX commands fully, he or she can immediately instruct the system on desired customizations. But the user who does not have a thorough knowledge of the commands can expect to be able to customize the system too, and to have to look a given syntactic construction up (or ask someone else) once at most, rather than again and again, as the interface will then adapt.

Admittedly, there is a lot of potential overhead in determining the meaning of an input from the user. The important thing to realize, however, is that "the price will have to be paid only once". This is in stark contrast to help systems and manuals that often have to be consulted over and over again. In fact, adaptivity is even better than that: the inadvertent use of the wrong command can be patched up once and for all. The DOS user who uses "dir" for UNIX "ls" can get the machine to adapt so that either the DOS command or the usual UNIX command will work. The UNIX user who often types "chmod foo 755" (rather than the correct "chmod 755 foo") because it seems more natural (more like "change the mode of foo to 755") will be able to use either order with impunity. If the user is willing to type more to use more English-like commands, the interface can be expected to adapt to "change the mode of foo to 755" or "change foo protection to 755", as well.

Dependence upon "clues" to the meaning is very much analogous to the way that people often extend their language capabilities, whether in their first natural language or a later natural or artificial language. As mentioned, the grammar and lexical and morphological routines of AIUI contain syntactic and semantic information like what is a file and what is an executable file. Although not included in the present design, a more sophisticated UNIX adaptive interface might want to check for unknown words in the UNIX on-line command manual to see if they are associated with given commands, etc. All of these sorts of heuristics should improve the dialogue, provided time cost is not too great (and that will vary with the speed of machines).

### 4.6. Generalization

Using this strategy, all possible user commands could eventually be mapped into system commands even without using structural descriptions and transformations thereof. That is, the strings could be mapped individually. But the process would be very slow. A mapping of a particular string would not tell much about the mapping of related strings. In order quickly to get to the point where the system can understand a great variety of user inputs, it is necessarily to map whole classes through the transformations. And, in fact, one wants to map classes that are as large as possible. In order to do this, it is necessary to generalize the results of a particular mapping.

Certain generalizations are fairly obvious. For instance, if a mapping exists from "run troff on aitrc.1" to "troff aitrc.1", then one should exist from "run nroff on aitrc.1" to "nroff aitrc.1" and from "run troff on aitrc.2" to "troff aitrc.2". One may get some overgeneralization by doing this, but overgeneralization is less of a problem in this sort of translation than it is where the same grammar is being used to produce strings of the language as a whole (i.e. of the user command language). It does not really hurt that the interface could handle inputs that the user would think ungrammatical.

There are other generalizations that are less obvious. For instance, if "run troff on aitrc.1" should be mapped to "troff aitrc.1", then "run troff -me on aitrc.1" should be mapped to "troff -me aitrc.1". If the grammar is to be efficient, then there are systematic reasons for wanting "run troff with me on aitrc.1" to be mapped to "troff (with me) aitrc.1", on the way to "troff -me aitrc.1", since in other contexts it is going to be necessary to transform "with me" to "-me". The design of the AIUI assumed that strong generalizations would be made.

### 4.7. "Worst Case" Structural Determination

It has been stressed that the type of adaptation that converges to an overall knowledge of the user's modes of interaction will only take place when the AIUI is provided with a corpus of inputs that it can classify structurally (parse) and with the meanings of those inputs. The structural classification cannot be expected to be perfect, however, so the system has "fall-back strategies" to use in the cases where it cannot parse the user input. These strategies use partial parses provided by a chart parser to create a hypothesized structure for the user input. Once it is transformed by a transformation derived after the meaning is determined, the hypothesized parse is added to the grammar. This will cause some variant structural assignments. In other words, the grammatical regularities developed in the grammar to that point will only be reflected in the portions that were matched by the partial parse. In the very worst case, this will cause a "flat" parse, dealing only with the lexical categories found for the given string. The degree to which the system will do this can be varied.

In investigations to date of the types of rules developed using these methods, the consequences of varying the structural assignments have not been great. The main problem caused by proliferating the grammar to deal with a lot of particular structures is the expansion of the grammar

and the UTD, and from a practical point of view that is undesirable because it will slow the adaptive process. Slowness (excessive reaction time) is a problem for a user interface, of course, and it will be necessary to see how much one could relax the restrictions without causing reaction time to become excessive. Another consequence could be slowed learning, since there can be less generalization.Obviously, the most desirable situation is where the grammar enables parsing of the user commands (which means that it has built into it, or has acquired, a syntactic knowledge, not necessarily a semantic knowledge adequate to the task). One would expect the typical situation to be somewhere between the worst and best cases.

**References**


Chomsky, Noam (1957). *Syntactic Structures*, Mouton and Co., 'S-Gravenhage.

Grinder, J. T. and Suzette H. Elgin (1973). *Guide to Transformational Grammar: History, Theory, Practice*, Holt, Rinehart & Winston, New York.

Lehman, J.F. (1989). Adaptive Self-Extending Natural Language Interfaces, PhD Dissertation, Carnegie-Mellon University, 1989.

Lehman, J.F. (1990a). Adaptive parsing: a general method for learning idiosyncratic grammars, *Proceedings of the 7th International Conference on Machine Learning*, Austin, TX.

Lehman, J.F. (1990b). Supporting linguistic consistency and idiosyncracy with an adaptive interface design, *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, Boston, MA.

Reeker, L.H. (1971). A problem solving theory of syntax acquisition, *Journal of Structural Learning*, vol. 2, no. 4, pp. 1-10.

Reeker, L.H. (1976). The Computational Study of Language Acquisition, *Advances in Computers, vol. 15* (Yovits and Rubinoff, eds.), 181-237, Academic Press, New York.

Reeker, L. H. (1980). Natural language programming and natural programming languages, *Australian Computer Journal*, vol. 12, no. 13, pp. 89-92.

Reeker, L. H. (1984).Adaptive natural language interfaces for computerized systems (Proposal for funding to AFOSR). Portions cited in Forsyth, R., and R. Rada. *Machine Learning: Applications in Expert Systems and Information Retrieval*, Ellis Horwood, 1986.

Reeker, L.H. (1988). Adaptive Individualized Interfaces for Computerized Processes, Report 1: Project Overview, September 1988; Report 2: Design Report (with L. W. Morrison), December 1988, Prepared for Applied Information Technologies Research Center, Columbus, Ohio.

Watt, W.C. (1968). Habitability, *American Documentation*, vol. 19.

# Explanation-Based Learning as a Tuning Tool for Large-Scale Natural Language Interfaces

**Christer Samuelsson and Manny Rayner**

Swedish Institute of Computer Science
Box 1263, S-164 28 KISTA, Sweden
christer@sics.se, manny@sics.se

## Introduction

> "When interacting in natural language it is easy to fall into assuming that the range of sentences that can be appropriately processed will approximate what would be understood by a human being with a similar collection of data. Since this is not true, the user ends up adapting to a collection of idioms --- fixed patterns that experience has shown will work."
>
> - Winograd and Flores , "Understanding Computers and Cognitions", 1985, p. 129.

Many users of natural language system tend to phrase themselves in the same way most of the time. The goal of this project has been to develop an optimization technique based on this observation: if one can speed up the analysis phase for the limited set of "typical phrases", one will save a great deal of computing time. The idea is that one can then bypass normal processing for most input sentences, instead using a set of specialized rules. This will be done paying the price of a small overhead when no special rule proves applicable. The set of special rules is extracted automatically, using explanation-based learning (EBL), from training sentences given by a user. EBL is a machine-learning technique related to chunking and macro-operator learning, that analyzes examples of successfully solved problems to find useful compositions of known rules. It is thus capable of improving a system's performance, but not of extending its knowledge.

By learning them from real user interaction, the set of special rules is tailored so as to capture the user's way of expressing himself. The hope is that a comparatively small set of language constructions will account for the majority of the sentences actually submitted to the system. Once the rules have been learned, it is important to store them in a way that minimizes the search for applicable ones at run-time, that is to index the learned rules so that quick access is guaranteed.

The project was begun in 1988, with the observation that the EBL method could readily be applied to clean logic grammars. In [Rayner 88], the fundamental ideas are described; some examples with toy grammars are presented, together with code for the EBL learning component, and a formal proof of its soundness. The learning component, or *generalizer*, has the form of a small Prolog meta-interpreter. In 1989 a series of more elaborate experiments were carried out on Fernando Pereira's CHAT-80 system, reported in [Rayner & Samuelsson 89]. These showed that it was possible to apply EBL to all steps of processing, from syntax up to the generation of logical forms; code for a "simplifier" was also presented, a module which performed a further partial evaluation of the learned rules to reduce their size. Most important, however, was a first version of an indexing mechanism, which made it possible to locate learned rules applicable to a given input sentence without performing a linear search. Indexing is performed by associating with each rule an atomic key, which encodes the lexical category information required for the words in input strings for which the rule is applicable.

In the last year, starting at the end of 1989, the method has been been successfully applied to two full-scale NL query systems, one of these being the well-known SRI Core Language Engine. This work is described in detail in [Rayner & Samuelsson 90] and [Samuelsson 91], and is summarized in the remainder of the present paper. The most important novelties are the use of a new indexing method based on a decision-tree approach, and the results of experiments carried out on a large test-corpus of 1663 sentences, derived from real user interaction. In these, the EBL-derived learned rules achieved a coverage of 90%, and total speed-up, measured over all sentences in the corpus, of a factor of 3 compared to normal processing. Median speed-up for sentences where a learned rule was applicable was approximately by a factor of 15, and median overhead on sentences where no learned rule was applicable was approximately 5%.

## References

[Rayner 88] Rayner, M., "Applying Explanation-Based Generalization to Natural Language Processing", *Proc. International Conference on Fifth Generation Systems*, Tokyo, 1988.

[Rayner & Samuelsson 89] Rayner, M. and Samuelsson, C., "Applying Explanation-Based Generalization to Natural Language Processing, part 2", SICS research report R89015, 1989.

[Rayner & Samuelsson 90] Rayner, M. and Samuelsson, C., "Using Explanation-Based Learning to Increase Performance in a Large-Scale NL Query System", *Proc. Speech and Natural LanguageWorkshop*, Hidden Valley, Pa., Morgan Kaufmann, 1990.

[Samuelsson 91] Samuelsson, C., *Using Explanation-Based Learning to Speed Up Natural Language Systems*, Licenciate Thesis, Royal Institute of Technology, Stockholm, 1991.

## Application to non-toy systems

The general architecture of the EBL module is the same in both of the large-scale systems to which it has been applied, and is illustrated schematically in the following two diagrams; the first shows the compile-time, and the second the run-time system.

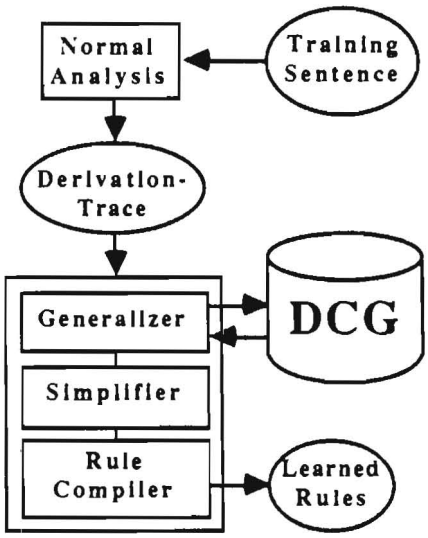



*Diagram 1, The compile-time component.*

The compile-time system contains three main components: the *generalizer*, which performs the actual extraction of learned rules; the *simplifier*, which attempts to reduce them in size by performing possible partial evaluations; and the *rule-compiler*, which adds indexing information to ensure quick access at run-time. The simplifier was not used in the Core Language Engine application.

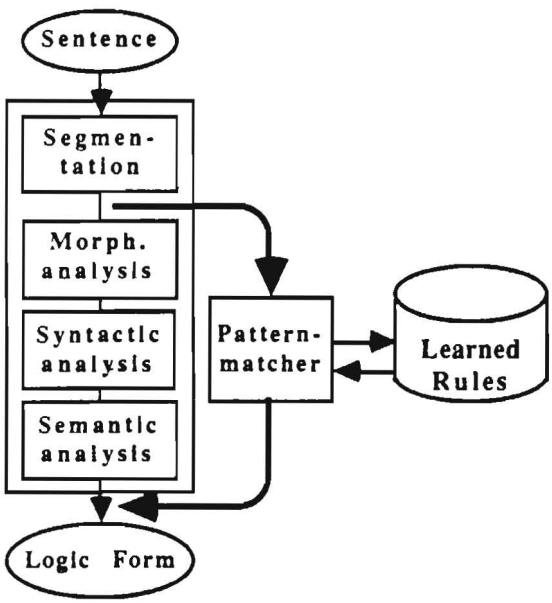



*Diagram 2, The run-time component.*

The run-time system consists of a single main component, the *pattern-matcher*, which attempts to bypass normal processing using the indexed set of learned rules produced by the compile-time system.

The detailed functionality of all components in the run-time and compile-time systems is described in [Samuelsson 91].

We now make a few remarks about problems specific to each of the two target systems, before presenting experimental results.

## The large-scale NL query interface prototype

Even though the large-scale NL query interface prototype had several characteristics that introducied problems when applying the EBL technique, it never the less proved possible to solve them. For this system the EBL module bypasses only syntactic analysis.

The main technical difficulties derived from the fact that our implementation of the EBL method requires the grammar to be reduced to a set of Horn-clauses. The two major hurdles with regard to the grammar formalism used are its non-standard treatment of features and movement: The basic feature operation is not unification, but priority merge. Movement is handled not by gap features, but rather by "non-restrictive" rules, in which more than one non-terminal can occur on the left-hand side of the rule as well as the right.

The problems connected with feature operations were solved by collecting all feature-manipulating predicates into the body of the learned rule and thus postpone all feature operations until run-time, when all feature values are properly instantiated. Doing this resulted in fairly large rule bodies, and a component for simplifying the learned rules was included.

The task of converting the the unrestricted grammar into a pure DCG form was performed by first representing the unrestricted grammar in Pereira's Extraposition Grammar (XG) format and then using an XG compiler to turn the grammar into pure Horn-clauses. Conceptually, the XG compiler turns the unrestricted grammar into a DCG, where each non-terminal is given an extra pair of arguments (the "extraposition list"), to pass around the additional left-hand constituents.

## The SRI Core Language Engine

EBL is very easy to apply to pure unification grammars such as the one provided with the SRI Core Language Engine. The rules extracted by the generalizer were sufficiently "clean" that there was no need to include a simplifier. For this system, we also extended the method by learning a set of rules that constructs words from word stems and affixes, i.e. that performs the task of morphological analysis.

## Results of experiments on the ATIS corpus

We now present a brief summary of experiments carried out at SRI Menlo Park in November, 1990. The EBL method was tested on the ATIS corpus, a large collection of sentences acquired by "Wizard of

Oz" methods, where subjects believed that they were interacting with a database through a real natural-language interface. It is therefore reasonable to suppose that these sentences are typical of real user interaction.

Two subsets were first selected randomly from the corpus, one of 1563 sentences for learning and one of 100 sentences for testing. The EBL method was applied to the learning set, resulting in the acquisition of 680 rules; these were then fed to the test set in increments of 20 rules at a time. After each increment, the test set was measured for rule coverage, average bypass time, and average performance gain. The coverage is defined as the fraction of a test corpus successfully handled by the EBL module. The bypass time is defined only for successful EBL look-ups and then as the processing times for the EBL module. The performance gain is simply the ratio of the total processing time for the test set using normal processing, divided by the total time using the learned rules.

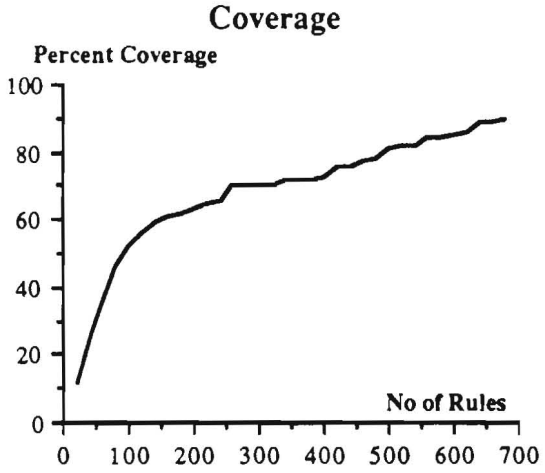



*Diagram 3, The coverage as a function of the number of learned rules.*

The coverage, as shown in diagram 3, swiftly rises to 60 percent for 150 learned rules and then increases more slowly reaching 90 percent at 680 rules.

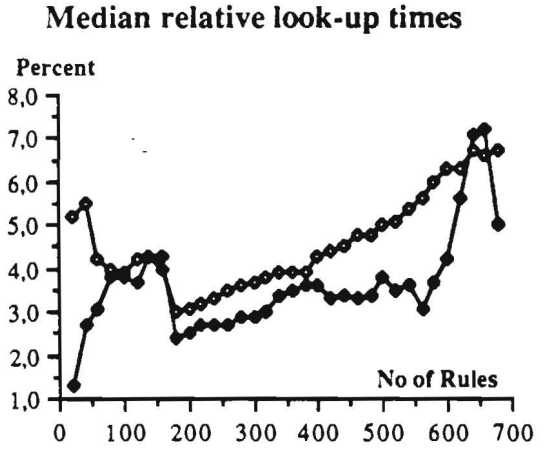



*Diagram 4, The median relative look-up time as a function of the number of learned rules.*

Diagram 4 shows the look-up time normalized by the time for corresponding normal processing. White diamonds indicate successful look-up, black failure. There is an increase with the number of learned rules. Though, the EBL look-up times are small compared to normal processing times - the median look-up times lie between 1 and 7.5 percent of normal processing time. With 680 rules, the median bypass time is 15 times less than that of normal processing and the median overhead is 5 percent.

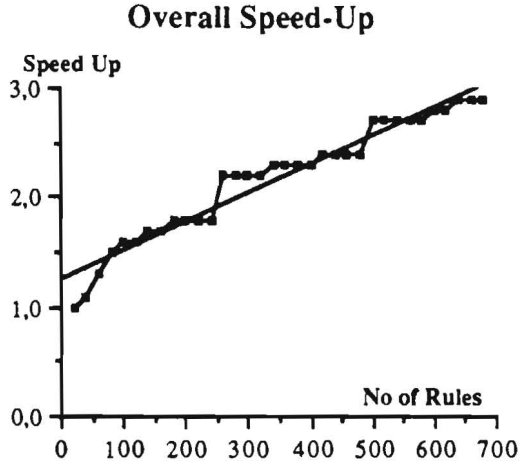



*Diagram 5, The overall speed-up as a function of the number of learned rules.*

This results in an overall speed-up increasing slightly sub-linearly in the number of learned rules. As can be seen in diagram 5, the system is twice as fast with 250 learned rules and with 680 rules the system runs three times faster.

It is our opinion that these experiments provide strong evidence to support the claim that EBL can substantially increase the performance of a natural-language interface under realistic conditions.

## Further directions

One obvious thing to do is to use the learned rules "backwards", that is for paraphrasing, by constructing an indexing scheme for logical forms.

Two interesting software engineering challenges are to integrate this scheme more closely with the target system to allow the normal analysis component to use partial results from the EBL module and vice versa, and to allow incremental adaption of the system by letting the learning component run as a background process.

Finally, we mention briefly a line of research that we have just begun to investigate, namely to incorporate the learned rules into a probabilistic language model of the kind used by speech recognition systems. Although our work to date is still only at a preliminary stage, it appears that this idea may potentially be very promising.

# Learning Simple Semantics by Self-Organization

J.C. Scholtes*
University of Amsterdam, The Netherlands

## Abstract

The recent neural network boom also inspired many researchers in the field of computational linguistics. Language seems well suited to be processed with the aid of neural-like computer architectures. Main technique used in various research projects is the Back-Propagation (BP) algorithm. On the one hand, known for its speed and relative mathematical simplicity. On the other hand, BP lacks psychological plausibility and self-organizing capabilities. To overcome the short-comings of supervised learning rules, the research carried out in this project evaluates the usability of self-organizing models in Natural Language Processing (NLP).

## Work done in Neural Networks and Natural Language Processing

Scholtes, J.C., Trends in Neurolinguistics, IEEE Symposium on Neural Networks, June 1990, Delft, Netherlands.

Scholtes, J.C., Using Extended Kohonen-Feature Maps in a Language-Acquisition Model, Submitted to the ACNN '91, 2-3 februari 1991, Sydney Australia.

Scholtes, J.C., Neurolinguistics, Computational Linguistics Project, CERVED S.p.A., Italy, 1990.

Henseler, J., Herik, H.J., Kerchhoffs, E.J.H., Scholtes, J.C. and Verhoest, C.R.J., Knowledge-Based Parallelism in Optical Character Recognition. Proceedings of the 1988 Summer Computer Simulation Conference, Seattle, 1988.

Herik, H.J., Scholtes, J.C. and Verhoest, C.R.J., The Design of an Parallel Knowledge-Based Optical Character Recognition System. Proceedings of the European Simulation Multiconference, Nice, June 1-3, 1988.

## Introduction and Current Research Interests

The author is member of the research staff of the Department of Computational Linguistics in the Faculty of Arts at the University of Amsterdam. After a masters in Computer Science at the Delft University of Technology and two years in the Royal Netherlands Navy, he joined the department in September 1989. His main task is the development and/or evaluation of connectionist and neural mechanisms for language processing. By doing so, self-organizing, statistical and selectionist techniques receive most attention. Other members of the research staff are specifically interested in data-oriented parsing and conceptual models in language-acquisition: other counterparts of the classical-language theories. By corporating closely together, new linguistic models are developed in various research contexts and directions. Future work might involve an even more neurophysical approach of cognitive issues, caused by the author's beliefs in the benefit of a synthesis of these two fields.

* The author can be reached at Bitnet: scholtes@alf.let.uva.nl, or by regular mail at: Dufaystraat 1, 1075 GR, Amsterdam, The Netherlands, or by fax at: +31 20 710793

## Background

Connectionism is often seen as the paradigmatic competitor of the symbolic tradition in artificial intelligence [Graubard, 1988]. Renewed interest in the field was mainly caused by the limitations of these symbolic methods and the practical problems occurring in the implementations of parallel algorithms [Herik et al., 1988], [Henseler et al., 1988]. Especially the property of connectionist systems to distribute knowledge with conservation of generality and integration was interesting for NLP research. Most popular in connectionist NLP is the BP algorithm [Rumelhart et al., 1986]. Although this algorithm started the 1980s neural bandwagon, it has some serious short-comings. First, the net can only learn input/output pairs, resulting in limited applicability. Second, automatic classification of raw data into various classes is impossible (leaving the need to predefine data categories on higher data abstractions, one of the main dilemmas of AI). Next, after the addition of new elements to the learning set, the entire set must be processed again. In other words, the model cannot adapt smoothly to a changing environment. Furthermore, the restricted architecture of BP nets (no inter-level connections) decreases complexity but increases neurological implausibility. More realistic are the self-organizing models, as proposed by Grossberg, Kohonen, Linsker and Von der Malsberg. These models can classify data automatically into non-predefined categories by forming a cortex-like map, and are capable to adapt slowly to an evolving environment, without the need to feed the entire learn-set over and over again. However, one of the main disadvantages of these models is the tremendous complexity. These models work fine in speech recognition and vision, but whenever one uses self-organizing models in NLP and other complicated processes, the complexity gets completely out of hand. Moreover, it is quite difficult to develop a self-organizing model, capable of doing more than sensor-based low-level pattern recognition. Although the results achieved in this context are still preliminary and not evaluated in depth yet, self-organizing systems might provide alternatives for some unrealistic assumptions in back-propagating neural nets.

## Introduction

Globally, the following self-organizing neurally inspired models are known from literature:

1. Grossberg's ART (much related to Von der Malsbergs' work),
2. Linsker's Implementations of Hebbian Rules,
3. Kohonen's Feature Maps, and
4. Reeke & Edelman's Neuronal Group Selection (NGS) theory.

All of them are based on variants of the Hebbian learning rule and the competitive-learning paradigm (the author realizes that there are many more variants on the above mentioned architectures, however, the models discussed here are best evaluated, making them more suitable to be used in NLP application research). Of all these models, Kohonen feature maps are most easy to simulate and are quite efficient for being self-organizing [Kohonen, 1984]. This restricted complexity is mainly caused by the facts that the model consists of one layer only, and the interneuronal connections are not learned (they are only used to implement lateral inhibition for the determination of the best match on the map). As a result, Kohonen maps are very efficient, but restricted in there usage: there are no built-in sequence handling mechanisms and it is impossible to implement hierarchical relations between objects formed on the map (it is quite irrelevant to let a map fire in the Kohonen formalism). [Grossberg, 1980] defines a model with two layers. Individual cells of the second layer correspond to the centers of clusters of input patterns. A neuron is connected to all the neurons in the opposite layer. According to a competitive learning rule, an adaptive model is obtained. Grossberg, like Kohonen, only learns the connections between layers. Connections within one layer are not changed. A more Hebbian way of learning can be found in [Malsberg, 1973] and [Linsker, 1988]. Hereby, there are no limitations to the interconnections of the model. All connections can be learned, as well connections between neurons of different layers as connections between neurons within one layer. Even more biologically inspired is the Neuronal Group Selection (NGS) theory as proposed in [Reeke et al., 1988] and [Edelman, 1989]. In this theory, a selectionist darwinistic approach is suggested to describe the process of group formation on the cortex map. Though these models are much more biologically likely than the other self-organizing models, they are heavily computational (if they can be simulated at all), and therefore not very popular in the already complex field of NLP applications. Moreover, the still developing ideas and the limited insight in the mathematical properties make these models less suited for the research of NLP applications.

Notwithstanding the fact that the Kohonen model is most restricted in its usage, the research carried out here, concentrated on extensions of this model. The main reason for this decision was just this restricted complexity and the mathematically provable convergation. Future work might concentrate on more complex models for the implementation of linguistic and psychological phenomena.

On the one hand, self-organizing systems can overcome some disadvantages of back-propagating neural nets. On the other hand, the theory of recurrent models is much less developed as it is in back propagation. Work by [Jordan, 1986], [Pineda, 1987] and [Williams et al., 1988] thoroughly analyzed recurrent back propagation. The impact on natural language processing of these techniques was demonstrated in [Elman, 1988], where the author showed the possibility to derive grammars from simple sentences. In [Servan-Schreiber et al., 1989], it was shown that the grammars derived by Elmans' model were finite state grammars. Although these grammars are definitely too simple to hold natural language completely, they have one interesting aspect in common with NLP; the appearance of a symbol (or word) in a string (or sentence) is determined by its precedents in that string. The addition of recurrent features in a self-organizing model completes the system with a implicit mechanism for temporal processing abilities, one of the important issues in natural language processing. In this context, work by [Allen, 1990] demonstrates even more potentiality of recurrent structures in connectionist NLP. Mainly for these reasons, the research carried out here, aims to use self-organizing models, which are able to process temporal sequences.

## Description of the Self-Organizing Models in NLP

The inability of Kohonen maps to process sequences by an implicit mechanism was encompassed in [Ritter et al., 1989] and [Ritter et al., 1990]. In general, it is of no use to process single words with a Kohonen map, because the formation of the structures on the map depends completely on the internal coding scheme. Therefore, sentences are presented to the system as a vector combination of words and their corresponding contextual structure. The structure formed on the map is related to the contextual position of words in sentences (called a Semantotopic Map of context). These semantics are just the ones, hardly derivable by logic and other symbolic techniques. The main disadvantage of this model is the inability to derive context by itself, mainly caused by the inability to process sequences and other temporal data. According to [Kohonen et al.,1981] temporal processing can be done by adding a buffering mechanism to a map. However, this mechanism does not support the automatic derivation of simple syntactic structures, used for further generalizations.

In [Kangas, 1990] a model is proposed, capable to process sequences within the Kohonen formalism (in fact, it is inspired by the second map used in Grossberg's ART, the functionality is almost the same). The model consists of two maps. The first one has a number of fibers, embodying the input vectors. For every neuron in this map, the measure of correlation is calculated by summing the difference between the input activities and the input weights. These values are combined into one vector, where every dimension represents a position on the first map. The second map uses these vectors as input, and learns in the same way as the first map. As a result, the second map holds the former activations of the first map, and is thus capable of forming a map representing sequences of input values (please note that this map holds no clustering information of the input values, but forms a map of activations of the first map).

As proposed in [Scholtes, 1990], the addition of recurrent fibers to the Kohonen model can provide the model with contextual sensitivity. To be more specific, the mechanism introduced by Kangas can be used for the derivation of context in the formation of a 'Semantotopic Map'. By feeding back the vectors of the second map, and concatenating this vector with the input vectors of the first map, the input of the first map results in a vector which has a symbol part and a automatically derived context part. As a result, the model can process simple sentences from scratch, and classify the objects in these sentences in semantic classes formed on the first map.

## Formal Description of the Model

Let m,u be vectors of dimension n in a map of $i$ neurons. $\mu_t$ holds the input vector and $m_t$ the input weights at time $t$. First the best match for a input vector is determined (i.e. the neuron which sensor
weights correspond best to the input values, conform to some mathematical distance measurement). According to the Kohonen rule, every learn cycle, the weights are adapted:

$$m_{t+1} = m_t + \beta_t \cdot (m_t - \mu_t) \qquad (1)$$

where $\beta_t = \varepsilon \cdot e^{-1/s^2}$ if a neuron is in the region of the most activated neuron, and $\beta_t = 0$ if the neuron is outside this region. $\varepsilon$ is a constant between 0.00 and 1.00 holding the learning speed, and $s$ is a decreasing function in time of the region size, forcing the system to converge.
The measure of correlation for each neuron $i$:: $y_{it}$ is then calculated conform:

$$y_{it} = 1.0 / ((m_t-\mu_t)^2+\delta) \qquad (2)$$

where $\delta$ is a very small constant, avoiding the system to divide by zero. Next, the firing rate is determined by:

$$y'_{it} = y_{it}^2 / Y_t \qquad (3)$$

$y_{it}$ represents the measure of activation of neuron $i$ at time $t$. $Y_t$ is the summation of all activations within one map. In addition, this output vector $y'_{it}$ can be averaged: $y''_t$ , so the system is less sensitive to noise:

$$y''_t = \omega \cdot y'_t + (1-\omega) \cdot y''_{t-1} \qquad (4)$$

This context part has dimension $i$, equal to the amount of neurons in the first layer. These two parts are concatenated, so $\mu_t = [\mu_{s1}, \mu_{s2}, ..., \mu_{sn}, \mu_{c1}, \mu_{c2}, ..., \mu_{ci}]$

In learning, the weights of the entire vector $m_t$ are adapted conform (1). The first map holds a spatial representation, the second one a temporal (caused by the averaging and the recurrent connections). Due to the combination of the two in a recurrent environment, the context of words is automatically stored in the second map, resulting in the completely unsupervised formation of a *Semantotopic Map* in the first layer.

## Results

The model is simulated by using the language C on a high-end PC and on a VAX 8250 mini computer. Three types of input were used: strings of 0 and 1, strings of characters, and simple sentences. The semantics of the first two input types are quite hard to define, so these strings were mainly used to show the ability of the system to classify objects in hypothetical semantic (or syntactic) classes. More interesting are the simple sentences, which were generated by the combination of a sentence body and some words which could be substituted into

$\omega$ holds the weigh factor, a value between 0.0 and 1.0, the higher $\omega$ the faster a new element is represented on this map, but the shorter the memory. The second map has dimension $i$ (each dimension represents the measure of correlation of a position on the first map). After normalization (3) and averaging (4) of the vector, the weights in the map are adapted in the same way, as the first map, so the second map holds the centers of activation occurred in the first map. As stated, the input vector consists of two parts: a symbolic part: $\mu_{st}$, representing a code pattern for a word, with dimension $n$. Second, there is a contextual part: $\mu_{ct}$, representing the activation of the map in the near past, with dimension $i$. Every cycle, the input values of the second map are copied into the context part of the first one.

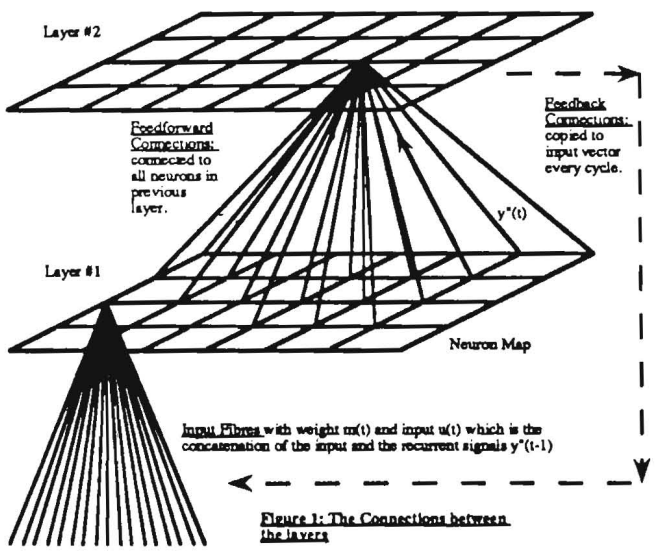


Figure 1: The Connections between the layers

the sentences. Most sentences contained three or four words. The model selects a number of sentences at random from a large amount of generated samples. First the elements in the samples are evaluated and a random code is assigned to all different words. Second, depending on the quantity of learn cycles, a random example is selected from the learn set and fed through the system. After 1000 up to 2000 cycles, a semantotopic map of context is formed of 50 small example sentences. The average map size was 25 neurons. The average input dimension 5. This small system already resulted in 30 dimensional input sensors.

To be more specific, input elements used for sentence generation were:

| | Elements | Sentences |
|---|---|---|
| 1. | John, Mary, Joe | 1/2/1 |
| 2. | Loves, Hates, Dislikes | 1/4/9 |
| 3. | Fast, Slow | 1/4/8 |
| 4. | Drinks | 1/10/6 |
| 5. | Cat, Dog, Fish, Horse | 1/10/7 |
| 6. | Meat, Bread | 1/4/7 |
| 7. | Much, Little | 1/4/7/8 |
| 8. | Beer, Wine | 5/10/6 |
| 9. | Well, Poorly | 5/10/7/6 |
| 10. | Eats | 5/4/7/8 |

Some sentences were: John loves Mary, Cats eats much bread, etc. After the learn cycles, semantic maps were formed were objects like *beer*, *water*, *meat* and *bread* were within a region. Another region was *dog*, *cat*, *John*, *Mary* and *Joe*. These maps are the same as the maps found in the work done by Ritter & Kohonen. Although, they added the context (a code for the sentence body) manually to the input vector. Here the input vectors are derived automatically by concatenating the symbol code with the recurrent context code.

In the simulations the model started with a large region (about the map size), which decreased slowly to 0.5. The learning rate (epsilon) was 0.75. The weight factor, W, averaging the measure of correlation from the first to the second map was 0.05. Simulation took up to 8 hours on the PC and up to more than 4 hours in batch on the VAX/VMS system for two maps of 50 neurons. Larger simulations were almost impossible, because the complexity increases exponentially with the map size (mainly caused by the amount of recurrent fibers needed).

## Discussion

The results presented are promising but preliminary. Important questions like the types of grammar the model can process, the maximum length of the sentences, and a better insight in the formation of semantic or syntactical maps have to be answered by future research. By now, only vague semantic groups were formed on the maps after a large amount of learn cycles. The exact reasons and conditions of map formations and thus a mathematical analysis must be worked out to define more thoroughly foundations for these phenomena.

As mentioned in the results, the complexity of the model increases exponentially with the map size. This growth might be limited by using the dimensions of the second layer as average representatives over regions of the first layer instead of one dimension for each neuron, which is definitely much too detailed. The size and type of these regions can be interesting material for further investigations. Beside the limitation of the models complexity, the investigation of different region types also has neurobiological reasons. Other variations to the model are the balance between the quantity of symbol- and context fibers, the rate of averaging over the second map, etc.
The importance of recurrent fibers for NLP might be clear. These connections made the automatic derivation of context possible. The lack of good definitions of recurrent mechanism in self-organizing systems leaves plenty space for further research towards other models. Linskers' work and the even more biologically inspired Neuronal Group Selection theory by Reeke & Edelman might be well suited to implement linguistic phenomena. Main problem with these models is complexity of simulations and the less developed foundations, making application research quite tricky. Various hybrid solutions tried to overcome the disadvantages of self-organizing models. A possible solution is to use a self-organizing feature map to discover the features in the learn set, and back-propagate between these maps to learn and generalize between input and output pairs (or between input patterns and regions on the map). The efficient back-propagation algorithm then limits the complexity and uses known mechanisms, like recurrent connections, to implement complex phenomena. More on these solutions can be found in [Hryceg et al., 1990] and [Gersho et al., 1990].

The advantage of the model discussed over back-propagating models might be clear. By generalizing over the context of small sentences, a semantical map is formed completely automatic without the usage of micro-features or predefined syntactical structures. The quality of the structural power and the semantical map is not evaluated in depth yet. Future research might provide us with more insights on these aspects. The advantages over symbolic natural language processing systems might be more than clear. Automatic derivation of structure and semantics by a system capable of generalizing over simple sentences which is robust to noisy input cannot be implemented in classical symbolic techniques easily.
Although Kohonens' self-organizing model is just an efficient statistical classifier, it is capable to derive semantical features of symbolic data, as long as this data is presented to the model in its proper context.
The same feature of neural nets can be seen in work carried out by [Miikulainen et al., 1988a], [Miikulainen et al., 1988b], [St. John et al., 1988a] and [St. John et al., 1988b], where generalization over context resulted in the automatically derivation of semantic (micro-) features. This ability of neural nets in general cannot be found in classical symbolic AI, without the addition of complex procedural modules.

## Conclusions

Problems seems to change nature when being represented in terms of temporal events, as a result, complex techniques like recursion in sequential processing can eventually be avoided. Nevertheless, the types of grammars, the length of the sequences and other properties of these models are quite unknown yet. Future research most provide a better insight in these aspects.

Self-organizing techniques can overcome some of the disadvantages of the back-propagation algorithm. Main problem with these self-organizing models is the quickly increasing complexity. Especially the addition of recurrent fibers enlarges the time required to process the input data. One might accept these disadvantages, because the limitation of back propagation (i.e. the need to learn input/output pairs, the prewiring of lateral inhibition, the definition of micro-features and the need to pass the entire learn-set again after addition of new elements) are even worse.

Recurrent self-organization is still in its early development. This is mainly caused by the limited knowledge of self-organization as a whole. Additional research can provide the insights needed here.

Although limited, a completely autonomous model for the derivation of context dependent semantics is developed (or in other words, semantical features are derived by generalizing over context). The exact properties are not known yet, but these semantics are just the semantics hard obtainable by logic and other commonly used semantic techniques. Just therefore the results are interesting enough to continue further research. Possible extensions might concern as well other, more powerful and complicated models, as more thoroughly defined examples in i.e. language acquisition as proposed in [Feldman et al., 1990] and [Weber et al., 1990].

## References

[Allen, 1990]: Allen, R.B. (1990). Connectionist Language Users. Technical Report. Bell Communications Research.

[Edelman, 1989]: Edelman, G.M. (1989). Neural Darwinism. Oxford University Press.

[Elman, 1988]: Elman, J.L. (1988). Finding Structure in Time. Technical Report CRL 8801, Center for the Research of Language, UCSD.

[Feldman et al., 1990]: Feldman, J.A., Lakoff, G., Stolcke A. and Weber, S.H. (1990). Miniature Language Acquisition: A Touchstone for Language Acquisition. TR-90-009, March 1990, ICSI Berkeley, CA.

[Gersho et al., 1990]: Gersho, M. and Reiter, R. (1990). Information Retrieval using a Hybrid Multi-Layer Neural

Network. Proceedings of the IJCNN, San Diego, June 17-21, 1990, Vol. 2, pp. 111-117.

[Graubard, 1988]: Graubard, S.R. (Editor) (1988). The Artificial Intelligence Debate. False Starts, Real Foundations. MIT Press.

[Grossberg, 1980]: Grossberg, S. (1980). How Does a Brain Build a Cognitive Code?. Psychological Review, Vol. 87, pp. 1-51.

[Hrycej, 1989]: Hrycej, T. (1989). Unsupervised Learning by Backward Inhibition. Proceedings of the 11th IJCAI, pp. 170-175.

[Jordan, 1986]: Jordan, M.I. (1986). Serial Order: A Parallel Distributed Processing Approach. Institute for Cognitive Science, Report #8604, University of California, San Diego.

[Kangas, 1990]: Kangas, J. (1990). Time-Delayed Self-Organizing Maps. Proceedings of the IJCNN, San Diego, June 17-21, 1990, Vol. 2, pp. 331-336.

[Kohonen et al., 1981]: Kohonen, T., Oja, E. and Lehtio, P. (1981). Storage and Processing of Information in Distributed Associative Memory Systems. In: Parallel Models of Associative Memory (G.E. Hinton & J.A. Anderson, Eds.), Lawrence Erlbaum.

[Kohonen, 1984]: Kohonen, T. (1984). Self-Organization and Associative Memory. Springer-Verlag.

[Linsker, 1988]: Linsker, R. (1988). Self-Organization in a Perceptual Network. IEEE Computer, Special Issue on Artificial Neural Systems, Vol. 21, nr. 3, pp. 105-117.

[Malsberg, 1973]: Malsberg, Chr. von der (1973). Self-Organization of rientation Sensitive Cells in the Striate Cortex. Kybernetik, Vol. 14, pp. 85-100.

[Miikkulainen et al., 1988a]: Miikkulainen, R. and Dyer, M.G. (1988). Encoding Input/Output Representations in Connectionist Cognitive Systems. Proceedings of the Connectionist Models Summer School, Carnegie Mellon University, pp. 347-356.

[Miikkulainen et al., 1988b]: Miikkulainen, R. and Dyer, M.G. (1988). Forming Global Representations with Extended Back Propagation. Proceedings of the 2nd IEEE ICNN, San Diego.

[Pineda, 1987]: Pineda, F.J. (1987). Generalization of Back Propagation to Recurrent Neural Networks. Psychological Review Letters, Vol. 59, nr. 19, pp. 2229-2232.

[Reeke et al., 1988]: Reeke, G.N. and Edelman, G.M. (1988). Real Brains and Artificial Intelligence. In: The AI Debate. False Starts, Real Foundations (R. Graubard, Editor), pp. 143-174. MIT Press, Cambridge.

[Ritter et al., 1989]: Ritter, H. and Kohonen, T. (1989). Self-Organizing Semantical Maps. Biological Cybernetics, Vol. 61, pp. 241.

[Ritter et al., 1990]: Ritter, H. and Kohonen, T. (1990). Learning "Semantotopic Maps" from Context. Proceedings of the IJCNN, Washington, 1990, Vol. 1, pp. 23-26.

[Rumelhart et al., 1986a]: Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986). Learning Internal Representations by Error Propagation. In: Parallel Distributed Processing. Vol. 1. (D.E. Rumelhart and J.L. McClelland, Eds.), pp. 318-362. MIT Press.

Scholtes, J.C., Using Extended Kohonen-Feature Maps in a Language-Acquisition Model, Submitted to the ACNN '91, 2-3 februari 1991, Sydney Australia.

[Servan-Schreiber et al., 1988]: Servan-Schreiber, D., Cleeremans, A. and McClelland, J.L. (1988). Encoding Sequential Structure in Simple Recurrent Networks. TR CMU-CS-88-183, Carnegie-Mellon University.

[St. John et al., 1988a]: St. John, M.F. and McClelland, J.L. (1988). Applying Contextual Constraints in Sentence Comprehension. Proceedings of the Connectionist Models Summer School, Carnegie Mellon University, pp. 338-346.

[St. John et al., 1988b]: St. John, M.F. and McClelland, J.L. (1988). Applying Contextual Constraints in Sequence Comprehension. Proceedings of the Cognitive Science Society, Montreal, 1988, pp. 26-35.

[Weber et al., 1990]: Weber, S.H. and Stolcke, A. (1990). L0: A Testbed for Miniature Language Acquisition. TR-90-010, July 1990, ICSI Berkeley.

[Williams et al., 1988]: Williams, R.J. and Zipser, D. (1988). A Learning Algorithm for Continually Running Fully Recurrent Networks. Technical Report ICS Report 8805, UCSD.

# How Do Children Learn to Recognize Ungrammatical Sentences?

Mallory Selfridge
Department of Computer Science and Engineering
The University of Connecticut
Storrs, CT 06269
mal@cse1.cse.uconn.edu

## Abstract

This paper addresses three questions on child language learning: "how do children learn to recognize ungrammatical sentences?", "how do children learn an infinite language from finite data?", and "how do children learn syntactic word classes?". This paper proposes answers based on side-effects of mechanisms used by the CHILD theory of child language learning. First, children learn to recognize ungrammatical sentences by learning a particular non-traditional positional syntax to guide understanding. Second, the language children learn is infinite because it is used to express an infinite meaning representation. Third, children don't learn syntactic word classes; the syntax they do learn does not use them. Current research is investigating how children infer the meaning of incompletely understood utterances, how they learn high-level knowledge structures by observation, and how the coverage of the non-traditional syntax can be increased.

## Research Interests in Child Language Learning

My overall interest in child language learning research is the development of a psychologically plausible theory that explains empirical data and supports a computer model of a child and parent in a micro-world that a user can "talk" to and which learns language and behaves in a manner similar to a child. Prior research involved development of the CHILD theory and computer model to address learning to understand simple imperatives, generate simple declaratives, accounting for certain psychological data. Current research focusses on three areas. First, I am applying the CHILD theory to traditional linguistic questions, developing extensions to the CHILD theory's representation of natural language syntax to more complex constructions, and considering further the problems of learning languages other than English. Second, I am developing a computer model, called TODDLER, of how children learn scripts and other high-level knowledge structures by observation, and how they use those structures for play within a case-based reasoning paradigm. Finally, I am investigating the problem of unifying TODDLER and CHILD to address the question of how children use learned high-level knowledge structures to infer the meaning of incompletely understood utterances during language learning.

## Representative Bibiography

Selfridge, M. and Schank, R. (1977). How to Learn/What to Learn. Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Cambridge, MA.

Selfridge, M. (1980). A Process Model of Language Acquisition. Ph.D. Dissertation. Computer Science Research Report #172, Yale University, New Haven, CT.

Selfridge, M. and Dickerson, D. (1985). Observational Learning by Computer. Proceedings of the AISB Conference on Artificial Intelligence and Education, Exeter, England.

Selfridge, M. (1986). A Computer Model of Child Language Acquisition. *Artificial Intelligence*, Vol 29, #2.

Selfridge, M. (1989). Papa, Why Does it Rain? or Building Expert Systems that Learn by Asking Questions. Presented at the Second Annual Symposium on Expert Systems in Business, Finance, and Accounting, Los Angeles, CA.

Selfridge, M. (1990). Why Do Adults Tell Stories? Why Do Children Play? For the Same Reason: Re-Indexing Old Cases Under New Generalizations. Proceedings of the 1990 Stanford Symposium on Case-Based Reasoning, Stanford University , Palo Alto, CA.

# How Do Children Learn to Recognize Ungrammatical Sentences?

Mallory Selfridge
Department of Computer Science and Engineering
The University of Connecticut
Storrs, CT 06269
mal@cse1.cse.uconn.edu

**Abstract**

This paper addresses three questions on child language learning: "how do children learn to recognize ungrammatical sentences?", "how do children learn an infinite language from finite data?", and "how do children learn syntactic word classes?". This paper proposes answers based on side-effects of mechanisms used by the CHILD theory of child language learning. First, children learn to recognize ungrammatical sentences by learning a particular non-traditional positional syntax to guide understanding. Second, the language children learn is infinite because it is used to express an infinite meaning representation. Third, children don't learn syntactic word classes; the syntax they do learn does not use them. Current research is investigating how children infer the meaning of incompletely understood utterances, how they learn high-level knowledge structures by observation, and how the coverage of the non-traditional syntax can be increased.

## 1.0 Introduction

This paper is concerned with three questions on child language learning: "how do children learn to recognize ungrammatical sentences?", "how do children learn an infinite language from finite data?", and "how do children learn syntactic word classes?". These questions arise from a linguistic account of language learning, which assumes that the primary process of language learning is learning a set of transformational grammar rules [1,3,4,5,12,15,16,17].

The CHILD theory of child language learning [25-33] is intended to account for a set of six psychological data, on the basis of psychologically plausible cognitive mechanisms, learning mechanisms, and natural language experiences. The CHILD theory was implemented and tested in a computer program, CHILD, whose behavior manifested the six data within a developmentally accurate progression. Since the CHILD theory explicitly proposes that children do not learn transformational grammars, it is important to ask whether the CHILD theory can provide answers to the three questions.

This paper presents answers to these questions that were developed within the CHILD theory as side-effects of mechanisms required to account for the six psychological data, and identifies further questions that are the subject of current research. It must be noted that the purpose of this paper is only to briefly summarize an approach to the problem of child language acquisition; it is beyond its scope to present detailed arguments in support of the positions expressed here.

## 2.0 Three Questions

The first question, that of how children learn to recognize ungrammatical sentences, can be restated in more empirical terms. For example, if a young child is asked whether the sentence "ball me the throw" sounds "silly" or "ok", chances are the child will respond "silly." Encouraged to "fix it up," the child may well generate "throw me the ball." Such behavior was reported by Gleitman, Gleitman, and Shipley [14] for children of two-and-a-half and five years. This behavior implies that by these ages children have acquired at least some ability to judge a sentence's grammaticality. Further, Gleitman et al. report that by age five, children's judgements increase in sophistication.

The second question, that of how children learn an infinite language, is commonly motivated by an observation: one can always add, say, "John said" to the front of a grammatical English sentence to produce a longer, equally grammatical, sentence. Observations such as this imply that natural language is infinite. However, as has also been widely observed, the language children are exposed to is both limited and finite. This apparent ability to learn an infinite language on the basis of limited and finite information suggests the question, how do children do this?

The third question, that of how children learn syntactic word classes, arises because any account of language learning which assumes that children learn transformational grammar must assume that children learn which class each word is a member of. The dependence on word classes by grammar-based approaches to natural language is the result of distributional analysis, which, basically, notes that if one word can be substituted for another in a sentence then the words can be assumed members of the same class.

Ideally, psychologically plausible answers to these questions should result in mechanisms which would behave as children do in those situations corresponding to the three questions. That is, a computer program based on such a theory should be capable of (1) learning to judge certain sentences "silly", (2) learning to generate sentences of indefinite length, and (3) learning to generate language showing substitutability, all within a psychologically plausible model of child language learning.

## 3.0 The CHILD Theory

The CHILD theory was intended to address a set of six data on child language learning during the ages of 10 months through five years, and proposed a set of psychologically plausible mechanisms to account for this data. The six data are:

- Comprehension precedes generation [2, 36]
- Vocabulary growth rate first increases, then decreases [36]
- Utterance length increases [10,11,13]

- Irregular words are regularized [13]
- Unlikely actives are initially misunderstood [35]
- Reversible passives are initially misunderstood [6,35].

These data were combined into a single composite eight stage developmental progression:

| | | |
|---|---|---|
| Stage 1 | (0;10) | Knows no language |
| Stage 2 | (1;0) | Learns 22 WPM (words per month), mean utterance length (MLU) is 1 word |
| Stage 3 | (1;6) | MLU is 2 words, uses present tense for both present and past |
| Stage 4 | (2;0) | Learns 30 WPM; MLU is 3, 4, and 5 words; uses a few irregular past tense words correctly; semantically unlikely actives misunderstood; passives understood using semantic likelihood |
| Stage 5 | (2;6) | Regularizes previously used past tense irregulars, learns 83 WPM |
| Stage 6 | (3;0) | Correctly uses regular and irregular past tenses, Learns 45 WPM |
| Stage 7 | (4;0) | All actives understood correctly, Reversible passives misunderstood |
| Stage 8 | (5;0) | Reversible passives understood correctly |

To address the six data and the progression, the CHILD theory proposed that children bring four cognitive capacities to language learning: knowledge of the world, basic mechanisms of language understanding and generation, the ability to mentally represent natural language meaning and syntax, and mechanisms to learn word meaning and syntax. When a child hears an utterance, he first understands it as well as possible using a preference-based semantic analyzer [9,33] that uses frame-based representations of word meaning [21] and syntactic positional knowledge of where in the utterance frame slot fillers are to be found. Then, the child uses knowledge of context to infer the complete meaning if necessary. Finally, he learns word meaning using concept learning techniques, and learns syntax by building disjunctive sets of positional syntactic features, using the predicates PRECEDES and FOLLOWS, to describe where slot fillers occurred in the input and then storing and updating those feature sets under the word whose meaning contained the slot.

The CHILD theory was implemented in a computer program, CHILD, that learned an English vocabulary involving active, passive, and prepositional phrase constructions via the eight stage developmental progression by being given experiences and language input that model those that children receive. Previous versions of CHILD have learned small subsets of Japanese, Spanish, and Serbo-Croatian [32].

## 4.0 How Do Children Learn to Recognize Ungrammatical Sentences?

An explanation of learning to recognize ungrammatical sentences requires an explanation of the ability to recognize ungrammatical sentences following learning. The CHILD theory proposes that a child recognizes that a sentence is ungrammatical, or "sounds silly", if any slot filler is in a position other than that predicted by the positional syntactic features associated with the word whose meaning had the slot. In such a case, some predicates would be false with respect to the correct slot filler, and a child could use those to generate an explanation as to why the sentence sounded silly. Since the CHILD theory assumes that the child has inferred the intended meaning of the sentence, he can "fix it up" by invoking his language generation mechanism on that meaning.

Given this account of recognizing ungrammatical sentences, the CHILD theory's answer to the question of how children learn to recognize ungrammatical sentences is straightforward: they do so by learning to understand, which involves learning the disjunctive sets of positional syntactic features that describe where slot fillers occurred in the input. Prior to learning syntax for a word, sentences in which fillers are out of position will not be recognized as ungrammatical; once a child learns syntax for a word, he can recognize when fillers for that word's meaning are out of position, and such sentences will be recognized as ungrammatical. This is demonstrated in the following summary of the performance of the CHILD program, in which CHILD learns to recognize ungrammatical sentences in a three stage progression, and generates corrections for those it recognizes as ungrammatical. In this example, CHILD knows meanings of "fed", "cereal", "Papa", and "Ethan", and assumes from world knowledge that Papa normally feeds Ethan.

**Stage A:** CHILD does not know syntax for "fed"
All sentences sound OK

| | |
|---|---|
| "Papa fed Ethan cereal" | understood correctly |
| "Ethan fed Papa cereal" | misunderstood |
| "Ethan was fed cereal by Papa " | understood correctly |
| "Papa was fed cereal by Ethan " | misunderstood |
| "cereal fed Papa Ethan " | |
| "was cereal Ethan fed Papa by" | |

**Stage B:** CHILD learns active syntax for "fed"
Actives sound OK; passives, probes sound silly

| | |
|---|---|
| "Papa fed Ethan cereal" | understood correctly |
| "Ethan fed Papa cereal" | understood correctly |
| "Ethan was fed cereal by Papa " | misunderstood, & sounds silly: "Ethan fed Papa cereal" |
| "Papa was fed cereal by Ethan " | misunderstood, & sounds silly: "Papa fed Ethan cereal" |
| "Cereal fed Papa Ethan " | sounds silly: "Ethan fed Papa cereal" |
| "Was cereal Ethan fed Papa by" | sounds silly: "Ethan fed Papa cereal" |

**Stage C:** CHILD learns passive syntax for "fed"
Actives, passives sound OK, probes sound silly

| | |
|---|---|
| "Papa fed child cereal" | understood correctly |
| "Ethan fed Papa cereal" | understood correctly |
| "Ethan was fed cereal by Papa " | understood correctly |
| "Papa was fed cereal by Ethan " | understood correctly |
| "Cereal fed Papa Ethan " | sounds silly: "Ethan fed Papa cereal" |
| "Was cereal Ethan fed Papa by" | sounds silly: "Ethan was fed cereal by Papa" |

## 5.0 How Do Children Learn an Infinite Language from Finite Data?

The question of how children learn an infinite language on the basis of limited and finite data has been addressed by a number of researchers [1,3,4,5,12,15,16,17], but their answers generally incorporate a grammar that includes recursive rules. The answer proposed by the CHILD theory is of necessity quite different, since the CHILD theory does not incorporate such a grammar with recursive rules. Instead, the CHILD theory's approach begins with the empirical phenomenon that is the basis of the observation that natural language is infinite, namely, that one can always take a sentence and make a longer sentence by, for example, adding "John said" to its front.

Given the phenomenon of "longer sentence generation", the question arises, how can a person do this? The CHILD theory proposes a three-part answer: first, the person understands the shorter sentence, and generates a meaning representation for it. Second, the person uses inference to embed the meaning of the shorter sentence within another concept. Third, the person expresses the resulting larger concept in natural language using his generation mechanism. Clearly, this process can be continued to produce sentences of arbitrary length. Given this, the CHILD theory's answer to the question of "longer sentence generation" is straightforward: they do so by learning word meanings and their associated syntax. Instead of relying on recursive grammar rules, the CHILD theory relies upon a knowledge representation that supports imbedding to an arbitrary degree and a non-recursive representation of syntax.

An important secondary question is why is the CHILD theory's answer a *preferable* to the traditional one? There are three reasons: first, it is simpler. Since all knowledge representation languages must support arbitrary embedding, accounts relying on grammars with recursive rules require both a complex knowledge representation language and a complex grammar, while the CHILD theory's account relies on a complex knowledge representation system and a relatively simpler representation of syntax. The second reason the CHILD theory's account is preferable is that it explains "longer sentence generation" as a side-effect of mechanisms independently required for language understanding, language generation and reasoning. Recursive grammar rules, on the other hand, are essentially a special-purpose mechanism designed specifically to explain "longer sentence generation." The third reason the CHILD theory's account is preferable is because it is psychologically plausible, since it directly addresses human behavior, and accounts for that behavior with mechanisms that are plausibly attributed to people.

## 6.0 How Do Children Learn Syntactic Word Classes?

The CHILD theory's answer to the question of how children learn syntactic word classes can be stated simply: children don't learn syntactic word classes. That is, the CHILD theory proposes that at no time during language acquisition do children learn that a given word is a member of a particular syntactic word class. There are a number of reasons for this, but the primary one is that none of the CHILD theory's mechanisms ever makes reference to the syntactic class of a word. The question arises, therefore, how the CHILD theory can account for those patterns of natural language that are used to motivate the need for syntactic word classes?

These patterns can be accounted for by the CHILD theory's standard language analysis and generation mechanisms. For example, to generate and understand the sentences "put the ball on the table" and "put the box on the table" requires only that the meanings of "box" and "table" be appropriate fillers for the OBJECT slot in the meaning of put, and that stored under "put" is the syntactic knowledge that one appropriate position for the OBJECT filler is FOLLOWING the meaning of "put" and PRECEDING the filler of the TO slot in the meaning of "put". No syntactic class information is ever associated with the words "box" or "ball", or any other word that can appear in this position.

A more complex example occurs in the sentences "John gave Mary a book" and "John gave Mary a kiss." Normally, "book" and "kiss" would be classified as nouns, in order to account for the fact that they can appear in the same position. In contrast, the CHILD theory proposes a different account of processing these two sentences. Specifically, in "John gave Mary a book", the meaning of "give" is the concept referring to the transfer of possession, and stored under "give" are sets of syntactic predicates specifying the positions of the fillers of that concept. In "John gave Mary a kiss", however, the CHILD theory proposes that understanding and generation focus on the word "kiss", and that the word "give" is used as a post-position function word marking the position of the ACTOR of the kiss. That is, the CHILD theory proposes that "John gave Mary a kiss" should be considered as another 'voice', like passive and active, and one that resembles Japanese and German in having the action word at the end of the sentence. Thus the CHILD theory suggests that the fact that "book" and "kiss" appear in the same position is, in some sense, fortuitous, and best explained by historical accident, rather than by placing "book" and "kiss" in the same syntactic word class.

## 7.0 Research Issues in Child Language Learning

This paper has summarized the answers provided by the CHILD theory to the questions of learning to recognize ungrammatical sentences, learning an infinite language from finite data, and learning syntactic word classes. Obviously, however, the CHILD theory is a long way from being a complete theory of child language learning. In particular, there are three important areas which require additional investigation. First, the CHILD theory proposes that child language learning depends upon the child's ability to infer the meaning of incompletely understood utterances, but its current inference mechanisms are not psychologically accurate. Current research is investigating the use of case-based reasoning [18,19,20,34] to model the child's ability to infer the meanings of incompletely understood utterances. Second, child language learning occurs while the child is also learning scripts and other high-level knowledge structures [22,23]. The

TODDLER project, currently underway, is modelling child learning of scripts and other high-level knowledge structures via learning by observation and play, and will be investigating the relationship between such learning and language learning. Finally, current research is addressing the issue of increasing the power of the CHILD theory's representation of syntax in order to account for additional empirical data on child language learning. Success in accounting for additional data would support the proposition stated at the beginning of this paper, that children do not learn transformational grammars [see also 7,8,24].

## References


[1] Anderson, J.R. (1981). A Theory of Language Acquisition Based On General Learning Principles, Proc. 7th IJCAI, Vancouver, Canada.

[2] Benedict, H., Language Comprehension in 10 to 16 Month-Old Infants. Thesis, Department of Psychology, Yale University, New Haven, CT , 1976.

[3] Baker, C.L. and McCarthy, J.J. (1981). *The Logical problem of Language Acquisition*. M.I.T. Press, Cambridge, Mass.

[4] Berwick, R. (1985), *The Acquisition of Syntactic Knowledge*, The M.I.T. Press, Cambridge, MA

[5] Berwick, R. and Weinberg, A., (1984), *The Grammatical Basis of Linguistic Performance -- Language Use and Acquisition*, The M.I.T. Press, Cambridge, MA

[6] Bever, T.G., The Cognitive Basis for Linguistic Structures. In J.Hayes (Ed.), *Cognition and the Development of Language* (Wiley, NY,19 70).

[7] Birnbaum, L. (1986). *Integrated Processes in Planning and Understanding*, Ph.D. Thesis, Research report no. 489, Department of Computer Science, Yale University, New Haven, CT

[8] Birnbaum, L. (1989). A Critical Look at the Foundations of Autonomous Syntactic Analysis, Proceedings of the Eleventh Annual Conference of the Cognitive Science Society, Ann Arbor, MI.

[9] Birnbaum, L., and Selfridge, M. (1981). Conceptual Analysis of Natural Language, in *Inside Computer Understanding: Five Programs plus Miniatures*. Schank R. and Riesbeck C.K. (eds.), Lawrence Erlbaum Associates, Hillsdale, NJ.

[10] Bloom, L., *One Word at a Time: the Use of Single Word Utterances Before Syntax* (Mouton, The Hague, 1973).

[11] Brown, R., *A First Language: The Early Stages* , Harvard University Press, Cambridge, MA

[12] Chomsky, N.(1965). *Aspects of the Theory of Syntax* M.I.T. Press, Cambridge, Mass.

[13] Clark, E.V. and Clark, H.H., *Psychology and Language* (Harcourt, Brace,Jovanovich, New York, NY,1978).

[14] Gleitman, L.R., Gleitman, H. and Shipley, E.F. (1972) The Emergence of the Child as Grammarian, *Cognition*, 1-2/3:1-164.

[15] Matthews, R.J. and Demopoulos, W. (1989). *Learnability and Linguistic Theory*, Kluwer Academic Publishers, Dordrecht, The Netherlands

[16] Pinker, S. (1979). Formal Models of Language Learning. *Cognition*, 7:217-283.

[17] Pinker, S. (1989). *Learnability and Cognition*, M.I.T. Press, Cambridge, MA

[18] Proceedings of the First Annual DARPA Case-Based Reasoning Workshop, (1988), Clearwater Beach, FL.

[19] Proceedings of the Second Annual DARPA Case-Based Reasoning Workshop, (1989), Pensacola Beach, FL.

[20] Riesbeck, C.K., and Schank, R.C. (1989). *Inside Case-based Reasoning*, Lawrence Erlbaum Associates, Hillsdale, NJ

[21] Schank, R. C., (1973). Identification of Conceptualizations Underlying Natural Language. In R. C. Schank and K. M. Colby (eds.) *Computer Models of Thought and Language*, W.H. Freeman and Co., San Fransisco.

[22] Schank, R.C. (1982). *Dynamic Memory: a Theory of Learning in Computers and People*. Cambridge University Press.

[23] Schank, R.C. (1989),. *That Reminds Me of a Story: A New Look at Real and Artificial Intelligence*, Book Manuscript.

[24] Schank, R.C. and Birnbaum, L. (1984), Memory, Meaning, and Syntax. In T.Bever, J.Carroll, and L.Miller, eds., *Talking Minds: the Study of Language in Cognitive Science*, The M.I.T. Press, Cambrdige, MA

[25] Selfridge, M. and Schank, R. (1977). How to Learn/What to Learn. Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Cambridge, MA.

[26] Selfridge, M. (1980). A Process Model of Language Acquisition. Ph.D. Dissertation. Computer Science Research Report #172, Yale University, New Haven, CT.

[27] Selfridge, M. (1981a) Why Do Children Say "Goed"? A Computer Model of Child Generation. Proceedings of the Third Annual Meeting of the Cognitive Science Society. Berkeley, CA

[28] Selfridge, M. (1981). A Computer Model of Child Language Learning. Proceedings of the First Annual Conference of the American Association for Artificial Intelligence, Stanford, CA.

[29] Selfridge, M. (1981b) A Computer Model of Child Language Acquisition. Proc. 7th Int. Joint Conf. on Artificial Intelligence. Vancouver, Canada

[30] Selfridge, M. (1982) Why Do Children Misunderstand Reversible Passives? The CHILD Program Learns to Understand Passive Sentences. Proceedings of the Third Annual Conference of the American Association for Artificial Intelligence, Pittsburg, PA.

[31] Selfridge, M. (1982). How Do Children Learn to Judge Grammaticality? A Psychologically Plausible Computer Model. Proceedings of the Fourth Annual Conference of the Cognitive Science Society, Ann Arbor, MI.

[32] Selfridge, M. (1985). Unpublished manuscript.

[33] Selfridge, M. (1986). A Computer Model of Child Language Acquisition. *Artificial Intelligence*, Vol 29,#2.

[34] Selfridge, M. and Cuthill, B.B. (1989). Retrieving Relevant Out-of-Context Cases: A Dynamic Memory Approach to Case-Based Reasoning. Proceedings of the 2nd Annual Case-Based Reasoning Workshop, Pensacola Beach, FL, May, 1989.

[35] Strohrer, H. and Nelson, K.E., The Young Child's Development of Sentence Comprehension: Influence of Event Probability, Nonverbal Context, Syntactic Form, and Strategies. *Child Development*, 45(1974), 567-576.

[36] Sutton-Smith, B., *Child Psychology*, (Appleton-Century-Crofts, NY, 1973).

# Dispelling Myths about Language Bootstrapping

and

# Naive Physics, Event Perception, Lexical Semantics and Language Acquisition

Jeffrey Mark Siskind[1]
M. I. T. Artificial Intelligence Laboratory
545 Technology Square, Room NE43-800b
Cambridge MA 02139
617/253-5659
internet: Qobi@AI.MIT.EDU

**Abstract**

Two competing theories have been proposed to explain how children begin acquiring language without any prior linguistic experience. The first, semantic bootstrapping, claims that children first acquire word meanings and then use this information to drive acquisition of syntax. The second, syntactic bootstrapping, claims the inverse, that children use some syntactic knowledge in figuring out what words mean. There are difficulties with both approaches. Semantic bootstrapping on one hand, requires a referential completeness assumption, that children possess a concrete understanding of the referent of each word before assigning a lexical category to that word and before formulating syntactic generalizations around those category assignments. Syntactic bootstrapping on the other hand, requires that children be able to recover the phrase boundaries of utterances, without the use of syntax, and be able to isolate verbs prior to knowing their meaning. Proponents of both theories argue their case by claiming that in principle, language acquisition is impossible without such assumptions. These papers attempt to refute such claims.

The first paper, an extension of work reported by Siskind (1990), presents a set of principles, implemented as an algorithm, that can simultaneously acquire syntactic parameters of $\overline{X}$ theory and a lexicon comprising both category and semantic information from a training corpus containing both linguistic and non-linguistic input. Before training, the algorithm does not possess a fixed grammar of the target language, nor any information, syntactic or semantic, about the words to be learned. No referential completeness assumption is made, nor does the algorithm require knowledge of the phrase structure of, or the lexical category of any word in, the linguistic input. The successful operation of the algorithm is demonstrated on training sessions both in English and in Japanese.

The methods described in the first paper require that the learner be able to attach a set of possible meanings to each linguistic utterance. The second paper focuses on how such a set can be derived from the non-linguistic context of an utterance, particularly the visual context. We describe a system, currently under construction, which observes a computer generated animation, constructed solely from line segments and circles, and given only continual updates of the positions, sizes and orientations of those line segments and circles at every frame, is able to construct a semantic representation of the objects in the animation, the changing spatial relations between them, and the events in which they are participating. Unlike some prior work in this area, the event perception mechanism we discuss functions independently of any linguistic input and does not require such input in order to correctly understand the visual information. Furthermore, unlike other prior work in this area, the event perception mechanism operates without the benefit of any object or event models. Instead, we incorporate into our theory a model of naive physics. The choice of which physical assumptions to incorporate into this model, namely substantiality, continuity, ground plane and gravity, is motivated by experimental evidence of the pre-linguistic knowledge possessed by infants. In the future, event perception will be tied to the language acquisition theory discussed in the first paper to yield a complete system for learning language from correlated visual and linguistic experience.

## References

[1] Jeffrey Mark Siskind. Acquiring core meanings of words, represented as Jackendoff-style conceptual structures, from correlated streams of linguistic and non-linguistic input. In *Proceedings of the* $28^{th}$ *Annual Meeting of the Association for Computational Linguistics*, pages 143–156, June 1990.

[1] This research is supported in part by a Presidential Young Investigator Award to Professor Robert C. Berwick under National Science Foundation Grant DCR-85552543, by a grant from the Siemens Corporation, and by the Kapor Family Foundation. Part of this research was performed while the author was supported by an AT&T Bell Laboratories Ph.D. scholarship and while the author was visiting Xerox PARC as a research intern and as a consultant.

# Dispelling Myths about Language Bootstrapping

## 1 Introduction

This paper addresses an issue in language acquisition which has become known as the *bootstrapping problem*. While in the later stages of language acquisition, children are assisted by previously acquired linguistic knowledge, how do children begin the language acquisition task without such knowledge? In particular, how do they assign syntactic categories and semantic representations to the words they hear as part of complete utterances?

Two competing theories have been proposed for solving this problem. The first, due to Grimshaw (1979, 1981) and Pinker (1984) states roughly that children first acquire the meanings of words and then use this information to derive the syntactic constraints of their language. This theory has become known as *semantic bootstrapping*. Semantic bootstrapping assumes that children first learn the meanings of words, by some unspecified mechanism. They then apply a default mapping to assign a syntactic category to each word based on its semantic category. In particular, THINGS are mapped to nouns and EVENTS are mapped to verbs. Such a default mapping has been termed a Canonical Structure Realization. Finally, syntactic rules are formed around these abstract syntactic categories which later generalize to cases where the words are not of the appropriate semantic class but are nonetheless of the appropriate syntactic class. For example, a child hearing the utterance *Fido barked* knows that *Fido* is a dog and that *bark* is an action and thus maps *Fido* to a noun and *bark* to a verb and forms the grammar rule $S \rightarrow N\ V$ as a template to account for the utterance. Elliott and Wexler (1986) propose a variant of of semantic bootstrapping which requires only that children map THINGS to nouns. Principles of universal grammar then assist the remainder of the bootstrapping process. Their scheme however, requires that children be able to recover the bracketed phrase structure of the utterances they hear solely from acoustic and prosodic information.

The second theory, due to Gleitman (1990), states roughly the converse: that children use syntactic information to acquire word meanings. This theory has become known as *syntactic bootstrapping*. Like Elliott and Wexler, Gleitman also assumes that children are able to recover the phrase structure of utterances from acoustic information alone, and that they use this phrase structure to derive the subcategorization frames associated with each verb. She then proposes that key elements of a verb's meaning can be derived solely from its subcategorization frame. For example, a child hearing the utterance *John told Mary that Bill left* will deduce that the verb *told* takes an NP complement and an $\overline{S}$ complement and thus is likely to be a verb of communication. Gleitman's method acquires word meanings from the utterances alone, without any reference to the non-linguistic context of the utterances. Brent (1990) has used Gleitman's method to learn components of word meaning by scanning large text corpora.

Proponents of both semantic and syntactic bootstrapping support their case primarily by arguing that *in principle* language acquisition must work their way as it is impossible to explain language acquisition without such an assumption. Grimshaw and Pinker (1990) attempt to refute Gleitman's claims by highlighting the fact that her theory does not offer a complete account of how verb meanings are acquired. They argue, that while her theory is in principle plausible, she has yet to prove that it is actually both a necessary and correct account of child language acquisition.

This paper takes a different approach. It argues that neither semantic nor syntactic bootstrapping are necessary to account for language bootstrapping. It does so by demonstrating an algorithm called DAVRA[1] which determines the syntactic category and meaning of words without relying on semantic or syntactic bootstrapping. It does not argue that semantic and syntactic bootstrapping are *wrong*, just that they are not *in principle necessary*. Likewise, there is no claim that DAVRA is a correct account of child language acquisition, simply that it is a plausible account. Determining which account is actually correct awaits further research.

DAVRA relies on a collection of syntactic and semantic principles, collectively termed Universal Grammar. Following the poverty of stimulus argument, DAVRA assumes that the language learner is innately endowed with a language faculty which incorporates the principles of Universal Grammar. These principles are summarized in Section 2. Unlike previous work (Siskind, 1990) which assumed a known fixed context free grammar prior to language acquisition, DAVRA uses instead a direct encoding of $\overline{X}$ theory including a capability for parametric variation in the language to be learned. DAVRA has successfully been applied both to English (Section 4) and Japanese (Section 5) examples, learning the correct $\overline{X}$ parameter settings for each. Several key points about this work deserve particular emphasis.

- A common assumption about language acquisition dating as far back as Locke (1690) is that children are presented with single word utterances, such as 'milk', in a context where it is clearly evident that 'milk' refers to milk. This is termed *referential completeness* and is a key assumption underlying semantic bootstrapping

[1] DAVRA or דברא is a make-believe Aramaic word for *word*.

(Bloom 1990). Heath (1983, 1989 p. 338) gives evidence that in some cultures, children are rarely presented with referentially complete stimuli yet they successfully learn language. DAVRA is not limited to single word utterances and furthermore allows the learner uncertainty in associating meanings with utterances.

- The learner starts out without knowing the meaning or syntactic category of *any* words in the linguistic input. This differs from some prior work (Granger 1977) which learns the syntactic category or meaning of words appearing in utterances with but a single unknown word, using the context of the remaining known words as a filter on possible syntactic category and meaning assignments for the unknown word. Furthermore, the learner starts out without knowing the $\overline{X}$ parameter settings of the language being learned. At the completion of the training session, the learner has acquired
  - a meaning for each word in the training session,
  - a syntactic category for each word in the training session and
  - the $\overline{X}$ parameter settings for the syntax of the language learned.

  This is true bootstrapping from nothing more than principles of Universal Grammar.
- We do not assume that the learner has access, via prosody, to the bracketed phrase structure of the linguistic input.

The key to the success of our paradigm is cross-situational learning. A number of prior approaches to language acquisition, in particular Elliott and Wexler (1986) and Lasnik (1989), attempt to demonstrate learning from a single utterance. We believe that in most situations, a single utterance does not offer enough constraint to uniquely determine either parameter settings or syntactic categories and meanings of words. Instead, we believe that the learner must find a lexicon and parameter settings which can simultaneously and consistently explain multiple utterances. Words that co-occur across multiple utterances are the keys which enable the learner to decipher the language acquisition puzzle. Note that this is not a form of distributional learning. In its classic form, distributional learning infers equivalence classes between word by observing two different words occurring in the same location within otherwise equivalent utterances. We make no such restriction on the form of the input nor do we require the learner to hypothesize any semantic similarity between a group of words to classify them as the same syntactic category.

## 2 A Linguistic Theory Supporting Language Acquisition

The linguistic theory incorporated into DAVRA is characterized by the following principles. We assume that the learner is innately endowed with a language faculty which operates according to these principles.

1. The learner is able to distinguish between linguistic and non-linguistic input. Normally, linguistic input is available on the auditory channel while non-linguistic input is available on the visual channel though this is not always the case. Whatever channels carry the linguistic and non-linguistic information (they may in fact be the same channel) the learner is able to separate and distinguish the linguistic from the non-linguistic information.

2. The learner is able to segment the linguistic input into sentences, to segment those sentences into words and to group different occurrences of the same word into the same equivalence class despite minor acoustic variation between occurrences.

3. The learner is equipped with a mechanism for representing meanings of individual words and entire utterances. All we require is that utterance meanings be represented by ground expressions in some calculus and that word meanings be represented by expressions in the same calculus, possibly containing variables. In this paper we arbitrarily take Jackendoff's (1983) conceptual structures as our meaning calculus. Thus the meaning of the utterance *The cup slid from John to Mary* would be $\mathrm{GO}(\mathbf{cup}, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{John}), \mathrm{TO}(\mathbf{Mary})])$ and the meaning of the word *slid* would be $\mathrm{GO}(x, [_{\mathrm{Path}}\ y, z])$. A companion paper (Siskind, 1991) discusses the inadequacy of this representation and proposes an alternate representation.

4. The learner is exposed to utterances in a single language. Each utterance the learner hears is grammatically correct in that language and the learner is able to associate each utterance with a set of possible meanings for that utterance. One of those possible meanings must actually be the correct meaning of the utterance. The learner's innate perceptual abilities combined with her naive theories of physics and psychology allow her to postulate plausible meanings for each utterance. Siskind (1991) proposes a mechanism for how this may be

done. Note that we do not require that the learner associate a *single* meaning with each utterance, rather that the learner postulate a set of plausible meanings, only one of which need be the actual meaning of the utterance. Future work will relax this constraint even further, allowing for some ungrammatical utterances or utterances for which none of the possible meanings associated with that utterance by the learner turn out to be correct.

5. The learner parses each input utterance according to the following variant of $\overline{X}$ theory:
   (a) Each nonterminal node in the parse tree has either one or two daughters.
   (b) The lexical categories are N, V, P and I.
   (c) Each lexical category X projects into the categories $X_{SPEC}$, $\overline{X}$ and XP.
   (d) Each utterance that the learner hears is of category IP.
   (e) $I_{SPEC}$ is processed as NP.
   (f) $\overline{I}$ is processed as VP. This differs somewhat from current linguistic theory and is done to simplify DAVRA. Future work will discuss modifications to DAVRA which handle $\overline{I}$ in accord with current linguistic theory.
   (g) The language the learner hears is either a *SPEC initial* language or a *SPEC final* language. If the language is SPEC initial then for every lexical category X, the language follows the rule

$$XP \rightarrow X_{SPEC}\ \overline{X}.$$

   If the language is SPEC final then for every lexical category X, the language follows the rule

$$XP \rightarrow \overline{X}\ X_{SPEC}.$$

   (h) The language the learner hears is either a *head initial* language or a *head final* language. If the language is head initial then for every lexical category X (except for I) the language follows the rule

$$\overline{X} \rightarrow \overline{X}\ YP$$

   for every lexical category Y. If the language is head final then for every lexical category X (except for I) the language follows the rule

$$\overline{X} \rightarrow YP\ \overline{X}$$

   for every lexical category Y. Furthermore, irrespective of whether the language is head initial or final, the language also follows the rule

$$\overline{X} \rightarrow X$$

   for every lexical category X (except for I).
   (i) The categories $X_{SPEC}$ (except for $I_{SPEC}$) and lexical categories X (except for I) are terminal.
6. A meaning is associated with each node in the parse tree. The meanings associated with terminals nodes are word meanings from the lexicon. The meaning associated with the root node is one of the meanings postulated for the utterance. The meanings associated with nonterminal nodes are related by the following *linking rule*.
   (a) If a node has a single daughter, then the meaning of the parent and the daughter are the same.
   (b) If a node $X$ has two daughters $Y$ and $Z$, then one of the daughters is called the *template* and the other is called the *argument*. We will call $X$ the *resultant*. The resultant meaning is derived from the template meaning by renaming the variables of the argument meaning so that they are distinct from those in the template meaning and then substituting the argument meaning for all occurrences of some variable in the template meaning. Alternatively, the argument meaning may be the distinguished symbol $\perp$, in which case the resultant meaning is the same as the template meaning.
7. Nodes of category $\overline{X}$ are templates while nodes of category $X_{SPEC}$ and XP are arguments.
8. Argument meanings must be variable-free.
9. A word cannot have a meaning which is just a variable.
10. A node of category XP cannot have $\perp$ as its meaning.

11. The following exceptions notwithstanding, any terminal can be non-overt, i.e. it may have no overt word associated with it.

    (a) The semantics of a node with no overt descendants must be $\perp$.

    (b) Nodes of category $\overline{X}$ must have at least one overt descendant.

12. The learner observes a *monosemy* constraint, i.e. the learner will assign each word at most one syntactic category and one meaning. Future work will relax this constraint. Other work in language acquisition often assumes a converse constraint that each distinct possible meaning be conveyed by at most one distinct word. Note that we do *not* require such a constraint ruling out synonyms.

Note that the above principles do not account for movement. While dealing with movement adds significant complexity to this system, there does not seem to be any reason why it could not be incorporated in a fashion analogous to the techniques used in this paper. This is a fruitful area for future research.

# 3 The Algorithm

DAVRA is written in a nondeterministic dialect of COMMON LISP known as SCREAMER (Siskind, 1991). DAVRA has been implemented and correctly processes the examples given in Sections 4 and 5. Due to length limitations on this paper, this section containing an annotated description of the main portion of the code had to be omitted. The full paper, as well as the complete code, are available from the author. Use of a nondeterministic dialect allows a straightforward and transparent encoding of the principles of Universal Grammar directly as statements in the program. While useful for pedagogical purposes, more efficient implementations are possible. Siskind (1990) discusses one such algorithm (called MAIMRA) for a linguistic theory which is similar to, though not identical to, the one presented in Section 2.

# 4 An English Example

Consider a scenario where the learner observes John rolling from a location near Mary to a location near Bill while hearing the utterance *John rolled.* The learner might hypothesize at least the following six potential meanings for that utterance since each of the following six events are subevents of the main event observed.

- John was near Mary (at the beginning of the main event).
- John was near Bill (at the end of the main event).
- John moved along some unspecified path.
- John moved along a path starting from a location near Mary.
- John moved along a path to a location near Bill.
- John moved along a path from a location near Mary to a location near Bill.

We presented DAVRA with a training session consisting of the nine utterances given in Figure 1. Each of the nine utterances was paired with between three and six possible meanings similar to those discussed above. These possible meanings were represented as Jackendovian conceptual structures.

Prior to the training session, DAVRA was not given any linguistic information other than the principles covered in Section 2. In particular, DAVRA was not given the $\overline{X}$ parameter settings for English, nor was DAVRA given the syntactic category or meaning of any of the words appearing in the training session. From this training session alone, DAVRA produces the following lexicon as output:

| |
|---|
| $\mathrm{BE}(\mathbf{person}_1, \mathrm{AT}(\mathbf{person}_3)) \vee \mathrm{BE}(\mathbf{person}_1, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{person}_1, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_1, \mathrm{FROM}(\mathbf{person}_3)) \vee$<br>$\mathrm{GO}(\mathbf{person}_1, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{person}_1, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_3), \mathrm{TO}(\mathbf{person}_2)])$<br>*John rolled.* |
| $\mathrm{BE}(\mathbf{person}_2, \mathrm{AT}(\mathbf{person}_3)) \vee \mathrm{BE}(\mathbf{person}_2, \mathrm{AT}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_2, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_2, \mathrm{FROM}(\mathbf{person}_3)) \vee$<br>$\mathrm{GO}(\mathbf{person}_2, \mathrm{TO}(\mathbf{person}_1)) \vee \mathrm{GO}(\mathbf{person}_2, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_3), \mathrm{TO}(\mathbf{person}_1)])$<br>*Mary rolled.* |
| $\mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_3, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{person}_2)])$<br>*Bill rolled.* |
| $\mathrm{BE}(\mathbf{object}_1, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{object}_1, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{object}_1, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{object}_1, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{object}_1, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{object}_1, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{person}_2)])$<br>*The cup rolled.* |
| $\mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_3, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{person}_2)])$<br>*Bill ran to Mary.* |
| $\mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_3, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{person}_2)])$<br>*Bill ran from John.* |
| $\mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{person}_3, \mathrm{AT}(\mathbf{object}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{person}_3, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{person}_3, \mathrm{TO}(\mathbf{object}_1)) \vee \mathrm{GO}(\mathbf{person}_3, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{object}_1)])$<br>*Bill ran to the cup.* |
| $\mathrm{BE}(\mathbf{object}_1, \mathrm{AT}(\mathbf{person}_1)) \vee \mathrm{BE}(\mathbf{object}_1, \mathrm{AT}(\mathbf{person}_2)) \vee$<br>$\mathrm{GO}(\mathbf{object}_1, [_{\mathrm{Path}}\,]) \vee \mathrm{GO}(\mathbf{object}_1, \mathrm{FROM}(\mathbf{person}_1)) \vee$<br>$\mathrm{GO}(\mathbf{object}_1, \mathrm{TO}(\mathbf{person}_2)) \vee \mathrm{GO}(\mathbf{object}_1, [_{\mathrm{Path}}\ \mathrm{FROM}(\mathbf{person}_1), \mathrm{TO}(\mathbf{person}_2)])$<br>*The cup slid from John to Mary.* |
| $\mathrm{ORIENT}(\mathbf{person}_1, \mathrm{TO}(\mathbf{person}_2)) \vee$<br>$\mathrm{ORIENT}(\mathbf{person}_2, \mathrm{TO}(\mathbf{person}_3)) \vee$<br>$\mathrm{ORIENT}(\mathbf{person}_3, \mathrm{TO}(\mathbf{person}_1))$<br>*John faced Mary.* |

Figure 1: An English training session presented to DAVRA

| Head Initial, SPEC Initial. | | |
|---|---|---|
| *John*: | [N] | $\mathbf{person_1}$ |
| *Mary*: | [N] | $\mathbf{person_2}$ |
| *Bill*: | [N] | $\mathbf{person_3}$ |
| *cup*: | [N] | $\mathbf{object_1}$ |
| *the*: | $[N_{SPEC}]$ | $\perp$ |
| *rolled*: | [V] | $GO(x, [_{Path}])$ |
| *ran*: | [V] | $GO(x, y)$ |
| *slid*: | [V] | $GO(x, [_{Path}\ y, z])$ |
| *faced*: | [V] | $ORIENT(x, TO(y))$ |
| *from*: | [N,V,P] | $FROM(x)$ |
| *to*: | [N,V,P] | $TO(x)$ |

Note that DAVRA has determined on the basis of the training session that English is both head initial and SPEC initial. Additionally, DAVRA has converged to a single meaning for each word in the training session, without referentially complete knowledge of the meaning of any of the training utterances. Furthermore, for all but the prepositions, DAVRA has determined a unique syntactic category for each word. The only uncertainty remaining after processing this session is whether *from* and *to* are nouns, verbs or prepositions. It is easy to see that DAVRA can never uniquely determine that an English preposition is in fact of category P since the principles incorporated into DAVRA allow nouns and verbs to appear anywhere prepositions can with the same semantic consequences. One must add further principles from Universal Grammar to DAVRA in order to allow her to distinguish prepositions. Incorporating a variant of case theory which states both that noun phrases must receive case and that nouns are not case assigners would allow DAVRA to determine that English prepositions could not be nouns since their complements would not receive case. Furthermore, noticing that English prepositions are never inflected would give indirect negative evidence (Lasnik, 1989) that they are not verbs. Adding such principles to DAVRA would remove any remaining uncertainty from the above training session.

# 5 A Japanese Example

MAIMRA, a predecessor of DAVRA discussed in Siskind (1990), is often criticized as being unrealistic due to its assumption of a fixed, built in grammar prior to lexical acquisition. DAVRA attempts to address this criticism by utilizing a parameterized variant of $\overline{X}$ theory instead of a fixed context free grammar, and acquiring the $\overline{X}$ parameter settings simultaneously with the lexicon from the same training session. To demonstrate the success of this approach, we translated the utterances of the training session from Figure 1 into Japanese, while leaving the non-linguistic input unchanged, and presented this new session to DAVRA. The translated utterances are given below:

*Taro ga korogashimashita.*
*Eriko ga korogashimashita.*
*Yasu ga korogashimashita.*
*Chawan ga korogashimashita.*
*Yasu ga Eriko ni hashirimashita.*
*Yasu ga Taro kara hashirimashita.*
*Yasu ga chawan ni hashirimashita.*
*Chawan ga Taro kara Eriko ni suberimashita.*
*Taro ga Eriko ni tachimukau.*

| Head Final, SPEC Initial. | | |
|---|---|---|
| *Taro*: | [N] | $\mathbf{person_1}$ |
| *Eriko*: | [N] | $\mathbf{person_2}$ |
| *Yasu*: | [N] | $\mathbf{person_3}$ |
| *chawan*: | [N] | $\mathbf{object_1}$ |
| *ga*: | $[V_{SPEC}]$ | $\perp$ |
| *korogashimashita*: | [V] | $GO(x, [_{Path}])$ |
| *hashirimashita*: | [V] | $GO(x, y)$ |
| *suberimashita*: | [V] | $GO(x, [_{Path}\ y, z])$ |
| *tachimukau*: | [V] | $ORIENT(x, y)$ |
| *kara*: | [N,V,P] | $FROM(x)$ |
| *ni*: | [N,V,P] | $TO(x)$ |

From these utterances, DAVRA produced the above lexicon as output. Again, DAVRA was able to uniquely determine the $\overline{X}$ parameter settings for Japanese, as well as unique meaning and syntactic category assignments for most words in the training session. Like before, the only uncertainty which DAVRA was unable to resolve was the assignment of category P to the words *kara* and *ni*. Methods similar to those discussed previously could remove this remaining uncertainty.

# 6 Conclusion

We emphasize that we have *not* demonstrated an algorithm that converges to parameter settings and a lexicon for all possible input of the form a child might encounter. While such a result is crucial for a complete account of child

language acquisition it is still beyond our current understanding. What we have done is to demonstrate, by way of a single example, how *in principle*, an algorithm can infer $\overline{X}$ parameter settings and a lexicon with neither semantic or syntactic bootstrapping assumptions. We also acknowledge that the linguistic theory incorporated into DAVRA has limited syntactic and semantic coverage. Nonetheless, we believe that the techniques discussed in this paper can be can be applied to build language acquisition models using more elaborate theories of syntax and semantics as such theories are developed.

**Acknowledgments**

The author would like to thank Linda Hershenson, Michael Caine and Yasuo Kagawa for help in translating the training session from English to Japanese.

# References

[1] Paul Bloom. *Semantic Structure and Language Development.* PhD thesis, Massachusetts Institute of Technology, September 1990.

[2] Michael Brent. Semantic classification of verbs from their syntactic contexts: Automated lexicography with implications for child language acquisition. In *Proceedings of the* $12^{th}$ *Annual Conference of the Cognitive Science Society*, 1990.

[3] W. Neil Elliott and Ken Wexler. A principle theory of categorial acquisition. In *NELS*, 1986.

[4] Lila Gleitman. The structural sources of verb meanings. *Language Acquisition*, 1(1):3–55, 1990.

[5] Richard H. Granger, Jr. FOUL-UP a program that figures out meanings of words from context. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pages 172–178, 1977.

[6] Jane Grimshaw. Complement selection and the lexicon. *Linguistic Inquiry*, 10:279–326, 1979.

[7] Jane Grimshaw. Form, function, and the language acquisition device. In C. L. Baker and J. J. McCarthy, editors, *The logical problem of language acquisition.* The M. I. T. Press, Cambridge, Massachusetts and London, England, 1981.

[8] Jane Grimshaw and Steven Pinker. Using syntax to deduce verb meanings. In *The Fifteenth Annual Boston University Conference on Language Development*, page 32, October 1990.

[9] Shirley Heath. *Ways with Words: Language, life and work in communities and classrooms.* Cambridge Press, 1983.

[10] Shirley Heath. The learner as cultural member. In Mabel L. Rice and R. L. Schiefelbusch, editors, *The Teachability of Language*, pages 333–350. Paul Brookes, Baltimore, 1989.

[11] Ray Jackendoff. *Semantics and Cognition.* The M. I. T. Press, Cambridge, Massachusetts and London, England, 1983.

[12] Howard Lasnik. On certain substitutes for negative data. In Robert J. Matthews and William Demopoulos, editors, *Learnability and Linguistic Theory*, pages 89–105. Kluwer Academic Publishers, Boston, 1989.

[13] John Locke. *An essay concerning human understanding.* Meridian Books, Cleveland, 1964. (Original work published 1690).

[14] Steven Pinker. *Language Learnability and Language Development.* Harvard University Press, Cambridge MA, 1984.

[15] Jeffrey Mark Siskind. Naive physics, event perception, lexical semantics and language acquisition. In *The AAAI Spring Symposium Workshop on Machine Learning of Natural Language and Ontology*, March 1991.

[16] Jeffrey Mark Siskind. *Screaming Yellow Zonkers.* Massachusetts Institute of Technology, 1991. forthcomming.

# Naive Physics, Event Perception, Lexical Semantics and Language Acquisition

## 1 Introduction

In a companion paper, Siskind (1991) argues that during the early bootstrapping stages of language acquisition, when children start out without knowing either syntax or the meanings of any words, children are aided in their task by hypothesizing a set of potential meanings for each utterance heard. For example, a child hearing the utterance *John entered the room* would look out into her environment and see John standing, walking, opening the door, being outside the room, and later being inside the room, along with many other possible events occurring in the environment unrelated to John. Hypothesizing that the utterance as a whole refers to one of those events aids the learner in figuring out what the individual words mean, as well as the syntactic categories of those words and the syntactic parameters of the language being learned. But how can a child hypothesize utterance meanings from visual perception? This is the topic addressed by this paper.

Since we want to understand how a child's perception of the world can aid the language acquisition task, we must look for evidence of what knowledge pre-linguistic children already possess prior to linguistic activity.[1] Spelke (1988) discusses habituation/dishabituation experiments which attempt to elucidate such knowledge. These experiments provide evidence that pre-linguistic children possess at least the following kinds of knowledge:

**substantiality:** the knowledge that objects take up space and cannot pass through one another,

**continuity:** the knowledge that an object appearing at point $A$ and then at point $B$ must have moved along a continuous path between those two points,

**gravity:** the knowledge that unsupported objects fall and

**ground plane:** the knowledge that the ground offers universal support for objects.

We refer to these collectively as pre-linguistic principles.

We are currently writing a program called Abigail, which attempts to incorporate such pre-linguistic knowledge into a simulated language learner to test the hypothesis that such knowledge can aid the language acquisition task. Abigail watches a computer animation constructed from line segments and circles. Along with that animation, Abigail receives a narration text describing the events occurring in the movie. The experimental paradigm of having a learner acquire new word meanings by watching a narrated movie has been explored by Rice (1990). In our case however, the learner is a machine rather than a child. Using techniques which incorporate the aforementioned pre-linguistic principles, Abigail analyzes the animation frame by frame and produces a semantic representation of the events occurring in that animation. The events of this semantic representation constitute the meanings hypothesized for utterances appearing in the narrative text. Siskind (1991) presents a learning algorithm which can utilize such a semantic representation to learn the syntactic categories and meanings of words. This paper focuses on how to produce this semantic representation from visual input using models of children's pre-linguistic knowledge.

Abigail lives in a microworld of animated movies. These movies contain objects which participate in events. The ontology of this microworld differs somewhat from that of our world. More importantly, however, the ontology of Abigail's world is similar enough to our world to model the pre-linguistic principles of substantiality, continuity, gravity and ground plane. A frame from one of Abigail's movies is shown in Figure 1. In this movie, the man walks to the table, picks up the ball, walks back and forth with it before putting it back on the table. Later, the woman repeats the same actions, and finally the man goes, picks up the ball and gives it to the woman who then puts it back on the table. Abigail's computational mechanisms are not specific to the particular objects and event in this movie. Unlike the system discussed by Badler (1975), Abigail does not possess any prior object or event models. Furthermore, the animation is generated by a program distinct from Abigail. Abigail has no access to the internal data structures of this animation program. Abigail observes only the positions, sizes, shapes and orientations of the line segments and circles comprising each animation frame. From this information, Abigail utilizes a theory based on the pre-linguistic principles of substantiality, continuity, gravity and ground plane to construct a semantic representation of the objects and events in the animation. Without any modification, Abigail can watch a different animation containing different objects participating in different events and still be able perform a semantic analysis to yield an appropriate representation of the objects and events in this new movie.

[1] This paper remains agnostic as to whether such pre-linguistic knowledge is innate or acquired during the early months of life.

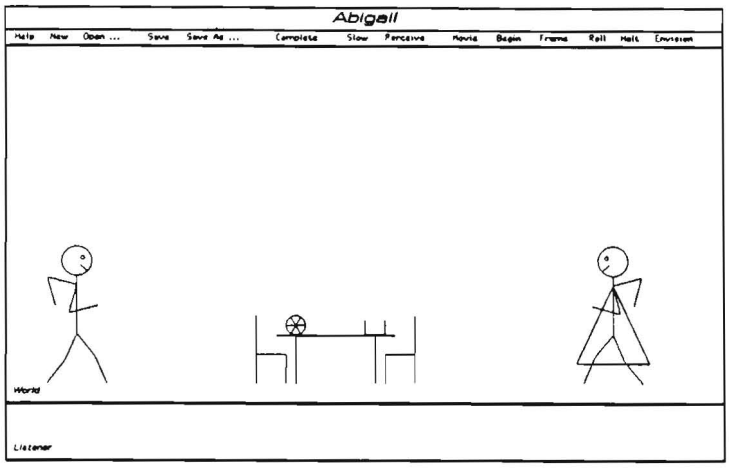



Figure 1: A frame from one of Abigail's movies.

# 2 The Theory

As mentioned previously, Abigail does not directly perceive objects such as people, tables and chairs. Instead Abigail perceives the figures, such as line segments and circles, out of which objects are constructed, and then interprets certain collections of figures as objects. In particular, Abigail understands that figures may be joined together. We denote a joint connecting figures $f$ and $g$ as $f \leftrightarrow g$. Such a joint can be described by three parameters: the displacement of the joint along the length of $f$, the displacement of the joint along the length of $g$ and the angle formed between $f$ and $g$. Any joint can be either rigid or flexible, independently along each of these three dimensions. A rigid joint parameter has some fixed value while a flexible joint parameter leaves its value unspecified. For technical reasons, we require that at least one of the displacement parameters of every joint be rigid. Any set of figures which are connected by joints will be interpreted as an object. Abigail does not directly perceive joints between figures. She infers those joints which are necessary to explain the unfolding animation according to the pre-linguistic principles. Furthermore, the set of joints and their parameter values need not be invariant for the duration of the movie. During the course of the movie, joints may change from rigid to flexible, or vice versa, and may even appear or disappear completely. This allows new objects to be built by combining old objects, old objects to be broken into parts and objects to be broken and then fixed again. Abigail, must continually maintain and update a joint model of the world to understand such construction and destruction events.

Abigail's microworld is nominally a two dimensional world. The figures that she perceives directly do not contain any depth information. Such a two dimensional world is not capable of supporting an interesting model of substantiality. The motion of objects in a two dimensional world which obeys substantiality is highly constrained. Nonetheless, when humans view the animation based on Figure 1 where the man walks from one side of the table to the other, they are not disturbed by the fact that in doing so, the man's figures overlap the table's figures. They never entertain the possibility that the man is walking through the table. Instead they assume that the man is walking either behind or in front of the table. In a similar fashion, Abigail attempts to reconstruct such depth information to explain the image and uphold the principle of substantiality. While not perceiving depth information directly, Abigail constructs a depth model which assigns certain figures constituting the image to the same layer and others to different layers. This model comprises a set of assertions of the form $layer(f) = layer(g)$, when figures $f$ and $g$ are known to be on the same layer, and $layer(f) \neq layer(g)$ when they are known not to be on the same layer. Only figures on the same layer must obey substantiality.

The layer model constitutes a partial third dimension. Abigail requires that at all times the layer model be a complete and consistent equivalence relation though not necessary total. Thus from $layer(f) = layer(g)$ and $layer(g) = layer(h)$ Abigail will infer $layer(f) = layer(h)$. Likewise, from $layer(f) = layer(g)$ and $layer(g) \neq layer(h)$ Abigail will infer $layer(f) \neq layer(h)$. However, for some pairs of layers, Abigail may not know whether or not they are on the same layer. Note that these layers are not ordered and in particular there is no notion of adjacent layers. Additionally, the assignment of figures to layers may change during the course of the movie. Thus Abigail must continually update the layer model both to maintain its internal consistency as well as to uphold substantiality judgments in the changing world.

The layer model consists of a list $(a_1, \ldots, a_n)$ of layer assertions. New assertions are always added to the front of this list. Whenever new assertions are added, we check the consistency of successively longer initial prefixes of the model. If the prefix $(a_1, \ldots, a_{i-1})$ is consistent but the prefix $(a_1, \ldots, a_i)$ is not, then the assertion $a_i$ is removed

from the model. This is repeated until the entire model is consistent.

How does Abigail apply the pre-linguistic principles to update both the layer and joint model? At every frame, Abigail looks for six types of evidence between every pair of figures $f$ and $g$.

1. Evidence that the assertion $layer(f) = layer(g)$ should be added to the model.
2. Evidence that the assertion $layer(f) \neq layer(g)$ should be added to the model.
3. Evidence that some parameter of the joint $f \leftrightarrow g$ should be demoted from rigid to flexible.
4. Evidence that an existing joint $f \leftrightarrow g$ should be removed from the model.
5. Evidence that some parameter of the joint $f \leftrightarrow g$ should be promoted from flexible to rigid.
6. Evidence that a new joint $f \leftrightarrow g$ should be added to the model.

Two forms of evidence can be used to infer case 1: support and collision. Whenever two figures touch and one would fall without being supported by the other, Abigail can infer that they are on the same layer. Likewise, if one figure moves toward another figure, touches it, and moves away from it according to the laws of physics, the apparent collision gives evidence that the two figures are on the same layer. Collision detection is not currently implemented in Abigail. In a similar fashion, there are two forms of evidence for case 2: overlap and exiting an apparent container. A direct observation that two figures overlap give clear evidence that they are on different layers. Furthermore, if one figure is initially surrounded by another figure and then moves so that it is no longer surrounded by that figure, the principles of continuity and substantiality imply that those two figures must be on different layers. Currently, only direct observation is implemented in Abigail. For case 3, an observation that the value of some rigid parameter of a joint has changed is evidence for demoting that parameter. An observation that two figures no longer intersect is evidence for case 4. Abigail currently does not implement any evidence for case 5. For case 6, Abigail infers a new joint whenever two figures touch and the two figures would cease to touch under the effect of gravity if they were not connected by a joint. In general, whenever Abigail hypothesizes new joints and same layer assertions to account for the stability of an object in the image, she attempts to hypothesize a minimal set of new joints and same layer assertions with same layer assertions taking priority over new joints when both offer the same explanatory power.

Central to the above process is a mechanism for determining support relationships between objects. Abigail uses a simulator for this purpose. This simulator takes the figures appearing in the current frame, along with a set of joints and layer assertions, and predicts how the image will change under the effect of gravity. This simulator is essentially a quantitative kinematic simulator that incorporates the pre-linguistic principles of substantiality, continuity, gravity and ground plane. It lacks any notion of dynamics, such as momentum, kinetic energy and friction. Nonetheless, it is adequate for determining the support relationships between objects, the same layer relationships between figures and the necessity of joints between figures.

Abigail continually performs such simulations every frame, hypothesizing what would happen in the world under different sets of joint and layer assertion assumptions. This has fairly strong psychological implications. For Abigail to be a plausible reflection of human perception, humans must be shown to be capable of performing such simulations and must also be shown to be performing them fairly regularly, albeit subconsciously. Freyd, Pantzer and Cheng (1988) gives evidence that humans perceive objects to displace slightly downward, as if they were falling, when support is removed from them.

Once Abigail has constructed the joint and layer model for each frame, and has collected connected figures into objects, she computes the following relations between those objects and the regions of space that they occupy:

$[i,j]exists(\alpha)$: Object $\alpha$ exists continually for frames $i$ through $j$.

$[i,j]contacts(\alpha,\beta)$: Object $\alpha$ touches and is on the same layer as object $\beta$ continually for frames $i$ through $j$.

$[i,j]joined(\alpha,\beta)$: For frames $i$ through $j$, objects $\alpha$ and $\beta$ are joined together by at least one joint connecting a figure from $\alpha$ to a figure from $\beta$.

$[i,j]supports(\alpha,\beta)$: For frames $i$ through $j$, object $\beta$ falls if the image is simulated without object $\alpha$ but object $\beta$ does not fall if the image is simulated with object $\alpha$.

$[i,j]supported(\alpha)$: For frames $i$ through $j$, object $\alpha$ does not fall when the image is simulated.

$[i,j]moving(\alpha)$: For every frame between $i$ and $j$, the position, size or orientation of some figure in object $\alpha$ has changed from the previous frame.

$[i,j]$*moving-root*$(\alpha)$: For every frame between $i$ and $j$, the position, size or orientation of some figure in the root of object $\alpha$ has changed from the previous frame. The root of an object is defined to be the subset of its figures which has the greatest mass and which is connected by joints which have not changed parameters since the previous frame.

$[i,j]$*translating*$(\alpha,p)$: Indicates that the center of mass of the root of object $\alpha$ is changing position for every frame between $i$ and $j$. The path $p$ is a trace of the movement of that center of mass.

$[i,j]$*rotating-clockwise*$(\alpha)$: The root of object $\alpha$ is rotating clockwise for every frame between $i$ and $j$.

$[i,j]$*rotating-counterclockwise*$(\alpha)$: The root of object $\alpha$ is rotating counterclockwise for every frame between $i$ and $j$.

$[i,j]$*rotating*$(\alpha)$: For frames $i$ through $j$, the root of object $\alpha$ is rotating either clockwise or counterclockwise.

$[i,j]$*place*$(\alpha,p)$: Object $\alpha$ occupies the region of space indicated by $p$ for frames $i$ through $j$.

*at*$(p,q)$: Points $p$ and $q$ are approximately coincident modulo a tolerance.

*in*$(p,q)$: Region $p$ is a subregion of region $q$.

*to*$(p,q)$: The ending point of path $p$ is approximately coincident with point $q$ modulo a tolerance.

*from*$(p,q)$: The starting point of path $p$ is approximately coincident with point $q$ modulo a tolerance.

*towards*$(p,q)$: Every point along path $p$ is closer to point $q$ than the previous point along that path.

*away-from*$(p,q)$: Every point along path $p$ is further away from point $q$ than the previous point along that path.

*up*$(p)$: The $y$-coordinate of every point along path $p$ is greater than the $y$-coordinate of the previous point along that path.

*down*$(p)$: The $y$-coordinate of every point along path $p$ is less than the $y$-coordinate of the previous point along that path.

# 3 An Example

The above relations are the primitives out of which semantic representations of events are constructed. Consider an event such as *John kicked the ball in the room*. This event could be represented as follows using the above primitives:

$$[t_1,t_2]translating(foot(\textbf{John}),p_1) \wedge [t_1,t_2]place(\textbf{ball},p_2) \wedge towards(p_1, center\text{-}of\text{-}mass(p_2)) \wedge$$
$$[t_2,t_2]contacts(foot(\textbf{John}),\textbf{ball}) \wedge [t_2,t_3]translating(\textbf{ball},p_3) \wedge$$
$$[t_3,t_4]place(\textbf{ball},p_4) \wedge [t_1,t_4]place(\textbf{room},p_5) \wedge in(p_4,p_5)$$

Each of the relations in the above expression can be derived from an animation of this event using the techniques described in this paper. A future paper will discuss how these relations are aggregated together to form the composite event description and how such an event description can be used by a language learner to learn the meanings of words in an utterance describing that event.

# References


[1] Norman I. Badler. Temporal scene analysis: Conceptual descriptions of object movements. Technical Report 80, University of Toronto Department of Computer Science, February 1975.

[2] Jennifer J. Freyd, Teresa M. Pantzer, and Jeannette L. Cheng. Representing statics as forces in equilibrium. *Journal of Experimental Psychology, General*, 117(4):395–407, December 1988.

[3] Mabel Rice. Preschoolers' QUIL: Quick incidental learning of words. In G. Conti-Ramsden and C. E. Snow, editors, *Childrens Language (Vol. 7)*, chapter 8, pages 171–195. Lawrence Erlbaum Associates, Hillsdale, NJ, 1990.

[4] Jeffrey Mark Siskind. Dispelling myths about language bootstrapping. In *The AAAI Spring Symposium Workshop on Machine Learning of Natural Language and Ontology*, March 1991.

[5] Elizabeth S. Spelke. The origins of physical knowledge. In L. Weiskrantz, editor, *Thought without Language*, chapter 7, pages 168–184. Clarendon Press, 1988.

# Explanation-Based Learning From Rule-Governed Features In Phonological Representations

Scott E. Stethem
Oracle Corporation
500 Oracle Parkway
Redwood Shores, CA 94065
sstethem@oracle.com

February 6, 1991

Abstract

An Explanation-Based Learning (EBL) system is used to learn macro-operators from rule-governed features in phonological representations. In the domain of generative phonology, EGGS-Phon follows the EBL machine learning paradigm to use existing domain knowledge in building explanations from examples. The system learns from relevant linguistic knowledge and produces macro-operators for handling vowel lengthening and consonant aspiration in English. EGGS-Phon's methodology is extended for languages which apply phonological rules in linear order.

# 1 Introduction

Explanation-based learning (EBL) takes existing knowledge (encoded as rules and facts) about a domain and "[constructs] an explanation for why a specific example is a member of a concept or why a specific combination of actions achieves a goal." [6] The explanation is typically in the form of a new rule, called a macro-operator. The system presented, EGGS-Phon, uses EGGS (Explanation Generalization using a Global Substitution) as its learning foundation [6]. EGGS-Phon is provided with basic concepts from generative phonology.

Many phonological features in an Underlying phonemic Representation (UR) are governed by rules, with each natural language having its own particular set of rules, rule orderings, and rule-governed features. EGGS-Phon's knowledge base specifies features about the phonemic inventory of languages, as well as their rule orderings.[1] EGGS-Phon takes a phonemic representation associated with a lexical entry (*i.e.*, the UR), and applies phonemic rules to produce a phonetic representation (which has a sufficient level of detail for pronunciation).[2] Examples of turning URs into Phonetic Representations (hereafter, PRs) are shown in figure 1. Two rule-governed features are illustrated:

- the feature *aspirated* (represented by "ʰ") for /p/, /t/, and /k/; and
- the feature *long* (represented by ":") for /æ/ and /o/.

EGGS-Phon uses the well-established observation that English unvoiced stop consonants (*i.e.*, /p/, /t/, and /k/) have their aspiration (*i.e.*, the puff of air accompanying its pronunciation) completely predicted by phonemic rules.[3]

| Lexical Item | Phonemic Representation | Phonetic Representation |
|---|---|---|
| *"pat"* | /p æ t/ | [pʰ æ t] |
| *"nab"* | /n æ b/ | [n æ: b] |
| *"cab"* | /k æ b/ | [kʰ æ: b] |
| *"potomac"* | /p ə t o m ɪ k/ | [pʰ ə tʰ o: m ɪ k] |

Figure 1: Examples of turning underlying lexical representations into phonetic representations

To generate the PR, EGGS-Phon:

- considers each phoneme of the UR (containing an ordered list of phonemes, enclosed in //) sequentially,
- , determines the values for all of the features of this phoneme which are governed by rules, and
- applies these rules.

At the end of this generation process (*i.e.*, after all phonemes are examined), the resulting phonological changes indicate the surface-level PR.

EGGS-Phon also orders rule applications, with each intermediate result serving as the phonological input to the next linearly ordered rule.

# 2 Motivation

Many Artificial Intelligence (AI) programs derive much of their inspiration from psychological plausibility. EGGS-Phon takes an analogous approach, employing linguistically plausible phenomena (*e.g.*, natural categories, rule-governed features, and ordered rules). By providing a computational framework for the linguist, the output of EGGS-Phon can reveal differences between the PR generated by the rules and the expected PR. The linguist may also evaluate the relevancy of learned macro-operators.

The EBL paradigm provides some advantages in representing phonological information. Linguists refer to "natural categories", notions which appear with great frequency across languages. Figure 2 illustrates some of these notions.

Natural categories can be kept as rules in EGGS, even as EGGS creates additional rules. A sample natural category rule is: *If* X *is consonantal and* X *is not sonorant, not continuant, not voiced, and not nasal,*

[1] Phonemes in this paper are written with symbols of the International Phonetic Alphabet (IPA) [2].

[2] Such rules and representations would be usable by various modules (*e.g.*, morphology, syntax, phonology) in a Natural Language Processing system.

[3] Aspirated stop consonants are in complementary distribution with their unaspirated counterparts. Thus, [pʰ æ t], not [p æ t], would be pronounced by native English speakers.

- positional information
  - word-initial
  - word-final
  - syllable-initial
  - syllable-final
- context-sensitive information
  - precedes-consonant
  - precedes-vowel
- phonological classes
  - stop-consonants (*e.g.*, /p/, /k/, /t/)
  - glides (*e.g.*, /h/, /w/, /y/)
  - affricates (*e.g.*, /č/, /t͡s/, /p͡f/)
  - fricatives (*e.g.*, /s/, /z/, /ʃ/, /θ/, /ʒ/)
  - front-vowels (*e.g.*, /i/, /e/, /æ/, /œ/)
  - open-unrounded vowels (*e.g.*, /a/, /ɑ/, /ɒ/).

Figure 2: Natural Categories

*then* X *has the feature stop.* Once exposed to examples (*i.e.*, lexical representations of words) needing features specified, EGGS generalizes from the example a new learned rule: *If* X *is a word-initial consonant, and* X *is consonantal and* X *is not sonorant, not continuant, not voiced, and not nasal, then* X *has the feature aspirated.*

# 3 System Description

EGGS-Phon stores phonological features about the consonants and vowels of the user-specified language in its knowledge rule base. EGGS-Phon utilizes the EGGS module (for generalizing explanations) as well as the DEDUCE module (for inferencing and proof generating).[4] As a rule-based system, EGGS-Phon works by iteratively applying concise "if-then" rules to a phonemic representation. An applicable rule modifies the UR by:

1. determining the feature's value (*i.e.*, presence or absence), and/or
2. inserting, deleting, or substituting phonemes.

Once EGGS-Phon has applied all rules, the resultant phonological representation is transformed into a surface-level PR. By learning a series of rules and generalizing over them into a single macro-operator, the system prepares itself for encountering similar examples in later input.

The phonological features of English used to demonstrate EGGS-Phon's feature processing are based on Fromkin and Rodman [4]. The linearly-ordered rules and sample words (including their intermediate phonological forms) of Tonkawa are from Kenstowicz and Kisseberth [5].

[4]Both modules are courtesy of R. Mooney.

## 3.1 Feature Processing

Languages have two types of features: immutable and rule-governed. EGGS-Phon takes the UR and returns a list of the features which must be changed to become a surface-level PR. More specifically, for each phoneme associated with the lexical entry, EGGS-Phon considers each rule-governed feature of the phoneme, with EGGS computing the present/absent value.

### 3.1.1 Rule-Governed Features

An example of this process is determining if the /k/ of /k a g/ (*"cog"*) is aspirated. When EGGS is asked to show that /k/ is aspirated, the system provides a proof that /k/ is aspirated because:

1. it is word-initial (and thus stressed syllable initial), and
2. it is a stop consonant.

This proof is generalized to become a macro-operator for later use.

### 3.1.2 Results

When EGGS-Phon was given the UR for *"potomac"*, it used facts from the knowledge base and rule applications to derive [pʰ], [tʰ], [o:] for the PR. Figure 3 shows the efforts involved in determining vowel length and consonant aspiration.[5]

From basic (user-specified) rules, EGGS learned two new more specialized rules: one for aspiration that takes place at the beginning of a word (as in the *"cog"* example), and another rule for aspiration that occurs at the beginning of a stressed syllable (such as the /t/ in *"potomac"*). Note that the learning generalizes the salient portions of the rule base's "if" conditions.

Figure 3 summarizes the findings on an English word *potomac*, which possesses three stop (unvoiced) consonants (with two being aspirated by different rules), and a vowel /o/ lengthened because it appears before a nasal (voiced) consonant. In both sections of the figure, when two numbers are listed in an entry:

- the first number indicates learning **without** interference (*i.e.*, learning when only user-defined rules exist for that predicate), whereas
- the second number indicates learning **with** interference (*i.e.*, learning when EGGS has defined a related macro-rule yet it does not apply to the phoneme and its current environment).

[5]Note that this example shows rule application as a one-pass process through the UR, with no dramatic alterations (such as deleting phonemes) to the UR.

**Before Learning**

| Phonemes for *potomac* | Rule Retrievals | Rules Tried | Answers Tried | Answers Made |
|---|---|---|---|---|
| aspirated /p/ | 14, 16 | 5, 6 | 12, 13 | 4, 4 |
| long /ə/ | 6, 10 | 2, 3 | 3, 6 | 0, 0 |
| aspirated /t/ | 16, 17 | 6, 7 | 13, 13 | 4, 4 |
| long /o/ | 6 | 2 | 5 | 2 |
| aspirated /m/ | 5, 7 | 3, 4 | 1, 2 | 0, 0 |
| long /ɪ/ | 6 | 2 | 3 | 0 |
| aspirated /k/ | 4, 5 | 3, 4 | 0, 0 | 0, 0 |

**After Learning**

| Phonemes for *potomac* | Rule Retrievals | Rules Tried | Answers Tried | Answers Made |
|---|---|---|---|---|
| aspirated /p/ | 9, 11 | 1, 2 | 8, 9 | 1, 1 |
| long /ə/ | 10 | 3 | 6 | 0 |
| aspirated /t/ | 10, 11 | 1, 2 | 9, 9 | 1, 1 |
| long /o/ | 5 | 1 | 4 | 1 |
| aspirated /m/ | 8 | 5 | 2 | 0 |
| long /ɪ/ | 10 | 3 | 6 | 0 |
| aspirated /k/ | 6 | 5 | 0 | 0 |

Figure 3: Learning Rule-Governed Features in *potomac*, where paired numbers **x**, **y** represent the totals for learning **without** interference and **with** interference, respectively.

In the After chart of figure 3, the first number indicates the benefit of learning macro-operators, benefits which start to decrease once there is learning with interference (as reflected in the second number). Note that "Answers Made" is non-zero only if the feature (*e.g.*, *aspirated*, *long*) applies.

Performance degradation (of serial systems) is a potential problem in EBL systems that learn macro-operators. This potential degradation can be seen in the attempt to discern that /m/ is not aspirated, where EGGS retrieved only five rules before learning but retrived eight rules after learning.

## 3.2 Rule Processing

An important notion of generative phonology is handling rules, with rule ordering being a key issue. While features are governed by rules, rules themselves are governed by ordering constraints.

### 3.2.1 Rule Ordering

The rule ordering portion of EGGS-Phon controls access to the inference mechanism (*i.e.*, EGGS and DEDUCE). Examples from Tonkawa [5] were chosen to investigate phonological rule orderings. Linguistic evidence for linear ordering is discussed by Kenstowicz and Kisseberth [5]. EGGS-Phon models Tonkawa with four linearly applied rules:

1. apocope — truncate a vowel if it ends a word,
2. elision — delete an element in a certain context,
3. truncation — delete the first of two consecutive vowels, and
4. vocalization — turn glides into vowels.

EGGS-Phon ran on six sample Tonkawa words. In all cases, the intermediate forms as well as the final forms were correctly generated: the rule ordering was never violated.

### 3.2.2 Results

The experiment verified that EGGS-Phon could allow rules to be ordered. While rules like truncation generalized, other rules (consisting solely of looking up facts) in the knowledge base did not generalize into macro-operators.

# 4 Directions for Future Research

While EGGS-Phon can support application of rules in linear order (*i.e.*, multiple rule firings), this system needs a more general mechanism than is currently implemented to process partially-ordered phonological rules.[6]

Anderson [1] also argues for a phonological component that supports cyclical application of rules, with rules being organized into sets in a hierarchy. Such notions are beyond the scope of the current implementation of EGGS-Phon, and would require a specialized inference mechanism.

EGGS-Phon has only been tested on intonational languages. The phonological representation currently implemented would have to be extended to handle tonal languages (*e.g.*, Mandarin Chinese).

"Noisy data" exists in all natural languages, where certain lexical entries (due to language change or historical accident) are exceptions to rules. Linguists call such "noisy" lexical items *suppletions*, and just list them in the data, with no further analysis. An avenue for enhancing EGGS-Phon would be to include examples with noisy data, and verify that no macro-operators would be learned from suppletions.

Since EGGS-Phon only used the EGGS system (and not the full GENESIS system), it does not have all the capabilities discussed in the later chapters of Mooney [6]. Specifically, EGGS lacks a schema learner. Only the GENESIS module provides *schema acquisition*: the ability to "[build] a schema describing plans for a wide variety of situations." [3] So, it

[6] Use of non-linear rule application is discussed in Mooney [6] but is not available in the EGGS module. Anderson provides linguistic evidence for partial ordering [1].

was not possible to devise an example that EGGS could solve *after* learning macro-operators that it could not have done *before* learning.

The most intriguing direction would be to incorporate "discovery learning" programs with EGGS-Phon. Using the basic linguistic knowledge base (consisting of natural categories and an inventory of phonological features) and a corpus of natural language examples, the "discovery learning" program could derive rules and generalizations that explain the data. EBL can then be used on these rules, both for testing their validity (whether given the same, subset, or different corpus of linguistic examples) and for finding the "short-cuts" (*i.e.*, macro-operators). However, the encoding strategy needed for this linguistic knowledge base to be usable by a discovery program (in making interesting new concepts) is not readily apparent, and would require further research.

# 5 Conclusion

EGGS-Phon demonstrates that explanation-based learning can be successfully applied to such linguistic domains as generative phonology. EGGS-Phon was able to learn rules for aspiration of stop consonants and for vowel-lengthening in English; its preliminary results of rule ordering were also encouraging. The functionality of EGGS-Phon (by facilitating tests of rules on natural language examples) could be of interest to computational linguists. With further enhancements (*e.g.*, to its inference mechanism), EGGS-Phon will have even greater applicability to generative phonology.

# 6 Bibliography


1. Anderson, Stephen R. *The Organization of Phonology.* Academic Press, Inc., 1974.

2. Catford, J.C. *A Practical Introduction to Phonetics.* Oxford University Press, 1988.

3. Ellman, Thomas. *Explanation-Based Learning: A Survey of Programs and Perspectives. Computing Surveys*, June 1989.

4. Fromkin, Victoria, and Rodman, Robert. *An Introduction to Language.* CBS College Publishing, Third Edition. 1983.

5. Kenstowicz, Michael, and Kisseberth, Charles. *Generative Phonology: Description and Theory.* Academic Press, Inc., 1979.

6. Mooney, Raymond. *A General Explanation-Based Learning Mechanism and Its Application to Narrative Understanding.* Technical Report AITR88-66, Artificial Intelligence Laboratory, University of Texas at Austin. Jan. 1988.


# 7 Acknowledgements


I am grateful to Steven Salzberg, Andrew Philpot, Jodie Kalikow, and others for reviewing earlier versions of this manuscript.

# Vector Space Grammars and the Acquisition of Syntactic Categories:

## Getting Connectionist and Traditional Models to Learn from Each Other

Andreas Stolcke

## Abstract

This papers describes a method for applying certain adaptive learning techniques usually found in connectionist systems to traditional symbolic grammatical descriptions. The approach is based on a generalization of context-free grammars in which discrete grammatical categories are replaced by elements from a continuous vector space, leading to the concept of Vector Space Grammars. Continuity of the representations, as well as differentiability with respect to rule application then enable use of learning techniques like competitive learning and error backpropagation. We show how this hybrid formalism can be used to learn grammar rules and category labels from phrase-bracketed positive and negative sample strings of a language.

Furthermore, since Vector Space Grammars are formally and conceptually derived from classical grammar formalisms, the results of learning can be analyzed and interpreted in terms of classical notions. In particular we show how the continuous rule representations learned can be analyzed to be quasi-isomorphic to classical context-free rules.

It is argued that generalization and extension of classical formalism to accommodate adaptive learning is a promising approach towards integrating traditional and connectionist methods.

## Author information

The work described in this paper is embedded in an ongoing research project at the International Computer Science Institute (ICSI). The $L_0$ project is an attempt to build a language learning system that covers only a miniature language and a limited artificial domain, but explores all levels of processing and representation in order to study problems arising from the interactions between such areas as knowledge representation, perceptual processing, and language. The present paper addresses a subproblem in $L_0$, namely syntax acquisition. Other researchers at ICSI and U.C. Berkeley involved in the project are Terry Regier (acquisition of spatial semantics), Adele Goldberg (cross-linguistic studies), Susan Weber (knowledge representation), George Lakoff (cognitive linguistics), and Jerome Feldman (director). This author's general interest are in linguistics and connectionist modeling of related phenomena.

Address: Andreas Stolcke, International Computer Science Institute, 1947 Center Street, Suite 600, Berkeley, CA 94704. E-mail: stolcke@icsi.berkeley.edu.

## Bibliography

Feldman, J. A., Lakoff, G., Stolcke, A., and Weber, S. H. (1990). Miniature language acquisition: A touchstone for cognitive science. Technical Report TR-90-009, International Computer Science Institute, Berkeley, Calif. Also appeared in the Proceedings of the 12th Annual Conference of the Cognitive Science Society, pp. 686–693.

Stolcke, A. (1989a). Processing unification-based grammars in a connectionist framework. In *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, pages 908–915, University of Michigan, Ann Arbor, Mich.

Stolcke, A. (1989b). Unification as constraint satisfaction in structured connectionist networks. *Neural Computation*, 1(4):559–567.

Stolcke, A. (1990). Learning feature-based semantics with simple recurrent networks. Technical Report TR-90-015, International Computer Science Institute, Berkeley, Calif.

# Vector Space Grammars and the Acquisition of Syntactic Categories:

## Getting Connectionist and Traditional Models to Learn from Each Other

Andreas Stolcke
Computer Science Division, UC Berkeley, and
International Computer Science Institute
stolcke@icsi.berkeley.edu

January 1990

## 1 Introduction

This paper describes work in progress aimed at exploring connectionist learning techniques for the construction of language acquisition devices. The main thrust of this work is to reconcile and capitalize on both the significant results in connectionist learning research and the body of linguistic knowledge as incorporated in standard high-level theories of language.

Connectionism, and especially Parallel Distributed Processing (PDP) has developed an array of models of learning systems (backpropagation, Boltzmann machines, competitive learning (Rumelhart et al., 1986b)), these models typically operate on representations at a rather low and unstructured level (unit activations, bit vectors, microfeatures) relative to the structures used in traditional linguistic descriptions (trees and graphs, case frames, grammar rules, stacks). This, of course, is no coincidence: the learning algorithms used, e.g., in Backprop Learning and Boltzmann machines are powerful and general precisely because they operate on simple and homogeneous representations. The simplicity of the representation allows a simple mathematical characterization and analysis, which in turn leads to (and justifies) the respective learning procedure (such as gradient decent and simulated annealing).

A second prerequisite for these connectionist learning algorithms is that representations be *continuous* in nature. Continuity of the representation space, with the added requirement that the performance measure be *differentiable* with respect to the representations, ensures that *adaptive learning* can take place, i.e., gradual adjustment towards a specified goal. Again, continuity and differentiability are typically not found in traditional linguistic descriptions, which for the most part are inherently discrete (Fuzzy Languages (Zadeh, 1972) are a notable exception).

The question that shapes up, then, is this: how can we harness the power of apparently powerful connectionist learning techniques without simply starting from scratch with respect to the linguistic insights gained and formulated within the existing theories of language. Put differently, how can we bridge the representational gap between these two fields so as to both extend the applicability of connectionist learning and add learning power to linguistic theories?

Assuming for a moment that we can be successful along these lines, an additional benefit becomes evident. If our connectionist representations are specifically designed to have a well-defined relation to existing theoretical constructs (such as linguistic rules and categories) we will increase the chance that the outcome of our learning procedure will not be just a collection of weights that apparently 'do the job'. Instead we can reinterpreted the solution found by the network in terms of the theoretical framework, even if that requires departing from some of the theoretical assumptions we started with.

This approach contrasts with some PDP research in which networks were trained on some linguistic task, and where the researcher *post hoc* tries to analyze the structures found and construct an adequate 'theory' of the networks internal behavior. The rational, of course, is to start out with an 'unbiased' network and to let it 'discover' the structure of the input as well as adequate internal representations. Invariably, however, the *post hoc* analysis has to refer to preformed concepts of language (Elman, 1988; Pollack, 1988; Elman, 1989). This is not surprising since many of those preformed concepts not only have a strong theoretical and empirical motivation, but are intuitive to some extent (like the fact that there are sentences and non-sentences, that verbs behave differently from nouns, etc).

## 2 Vector Space Grammars

The work reported here is a specific example of how traditional linguistic concepts might be combined successfully with adaptive learning techniques to result in a framework within which certain aspects of language and grammar can be learned.

We have developed a generalization of traditional context-free grammars (CFGs), called *Vector Space Grammars* (VSGs). VSG rules have the same format as standard CFG rules in Chomsky Normal Form (CNF), namely nonterminal productions of the form

$$X \rightarrow Y\ Z \tag{1}$$

and lexical (terminal) rules

$$X \rightarrow \mathrm{a} \tag{2}$$

to derive strings of a language. Whereas in traditional grammars categories ($X$, $Y$, $Z$) are symbols in a space with a

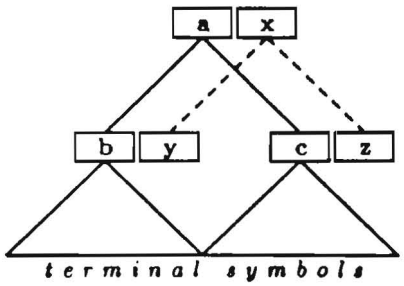



Figure 1: Vectors involved in VSG rule application. The new root vector **a** is a function of the subtree root vectors **b** and **c** and the vectors in the rule $\mathbf{x} \rightarrow \mathbf{y}\ \mathbf{z}$, e.g., $\mathbf{a} = (\mathbf{b}\cdot\mathbf{y})(\mathbf{c}\cdot\mathbf{z})\mathbf{x}$.

binary metric (equality/nonequality), VSG uses *vectors* as nonterminal categories. This gives a continuous metric on the category space, thus fulfilling one of the prerequisites for an adaptive learning mechanism. Terminals (words) in VSG are still unanalyzed atomic entities, and strings of terminals form the domain in which a language is defined.

A standard non-terminal rule maps two specific symbolic categories into a third symbolic category (the left-hand side of the rule). Similarly, a VSG rule maps two vectors onto a third. From a bottom-up parsing point of view, a traditional CFG rule is applicable if and only if its two right-hand side categories match exactly two other categories (roots of partial parses). In VSG, rule applicability becomes a graded notion, and every rule will be applicable to every two categories to some extent. However, the formalism is designed such that well-matching rules give a 'high' output, and poorly matching rules result in a vector close to the zero vector. This is accomplished by the following 'activation function' for VSG rules. Let $\mathbf{x} \rightarrow \mathbf{y}\ \mathbf{z}$ be the rule applied to two categories **b** and **c** (we use bold letters to denote vector quantities). Then the category resulting from the rule application is defined as

$$\mathbf{a} = (\mathbf{b}\cdot\mathbf{y})(\mathbf{c}\cdot\mathbf{z})\mathbf{x} \qquad (3)$$

where $\cdot$ denotes the inner product of the vector space. The two inner products on the right express the match of categories, and since the right-hand side terms in a context-free rule work conjunctively (all have to match), the values are multiplied. Choosing the inner product as the measure of matching partly determines the structure of the category space: categories will behave differently to the extent that they are orthogonal. The elements involved in rule application are depicted schematically in figure 1.

It can be shown that traditional CFGs and their way of rule application is a special case of VSG rule application. Roughly speaking, each dimension in the category space corresponds to a non-terminal in a traditional grammar.

*Acceptance of strings* by a grammar can be defined analogously to traditional grammars, although acceptance becomes a non-discrete function (similar to Fuzzy Languages). Since these definitions are not directly relevant to rule formation we will omit them here and turn immediately to the learning algorithm (see (Stolcke, tion) for details).

# 3 Learning with Vector Space Grammars

The problem of learning to parse strings of a language can be broken down into two subproblems: finding the structure of the parse tree, and assigning category labels to the nodes in the tree. There are indications that the two problems might in fact be handled separately.

Morgan et al. have shown that this assumption can be justified from at least three perspectives. Firstly, across natural languages there is a variety of cues present in the surface structure of language (both intra-sententially and cross-sententially) which correlate well with phrase boundaries and would therefore form a suitable basis for phrase structure extraction prior to grammar learning. Secondly, it can be shown that at least adults actually depend on these cues when taught artificial languages (Morgan et al., 1987; Morgan et al., 1989). Finally, learnability arguments show that the absence of such prestructuring in grammar learning would require unrealistically large amounts of processing capacity and input samples (Morgan, 1986) for learning to be successful on theoretical grounds.

In the following we will discuss how the category system and the rules for a language can be learned within the formal framework provided by VSG, given positive and negative instances of the language along with their phrase structure boundaries.

It should be pointed out at this stage that the overall algorithm about to be described is not connectionist in the sense that, for every aspect of its operation, a neurally plausible implementation can be given. In particular, structures will be created dynamically throughout the algorithm, something for which no elegant connectionist mechanism is known so far. However, the structures themselves (VSG rules), as well as the operations involved in the application of individual rules and in the learning procedure, *are* implementable with mainstream connectionist hardware.

Two global parameters of the system are the dimension of the category space and the number of rules to be used. These parameters should be set 'large enough' for a given language, and have an effect similar to the number of hidden units in a backpropagation network. With too little resources, the system will not converge on a solution, and with too many degrees of freedom the solution might be redundant and not express certain generalizations about the input.

At the outset of learning, then, a fixed number of non-terminal rule 'templates' of the form (1) (with a given vector space dimension) are allocated. Additionally, for each terminal symbol, a rule of the form (2) is created. All category vectors, in all rules, are set to random unit-length vectors.

Given a sample string from the language and a parse tree skeleton, we construct a labeled parse tree from the current set of rules. To assign a category vector to a node, the rule whose right-hand side represents the best match for the child node categories is selected and equation (3) is used to compute the output category for that node. 'Best match' is defined according to the same inner product metric as used in equation (3), i.e., using the value $(\mathbf{b}\cdot\mathbf{y})(\mathbf{c}\cdot\mathbf{z})$. Only the rules selected at some node will later participate in the learning process, and since only the currently best rules get selected the whole process strongly resembles the method of *competitive learning* (Rumelhart and Zipser, 1985).

By working from the terminal nodes to the root we arrive at a category label for the entire string. If the training sample is a positive instance of the language we know what the target category for the parse should be: the sentence category 'S'. Without loss of generality we can fix S throughout training to be a particular vector, e.g., the unit vector $(1, 0, \ldots, 0)$.

The second idea adapted from connectionist learning methods is that of *error backpropagation* (Rumelhart et al., 1986a). At the root node we can immediately compute an error term for the discrepancy between the desired output and the actual output. For positive examples this is just the difference between S and the root category, for negative examples we compute an error term which tends to make the output category and S orthogonal. A recursive procedure (based on the chain rule) can then compute the derivative of that error with respect to every category vector occurring in some rule (left of right-hand side) applied somewhere in the tree. The details of the computation of derivatives can be found in the appendix.

Derivatives for each category vector are then added up and multiplied by some constant (the 'learning rate') to give the adjustment to be applied to that category. All rules are updated accordingly, all categories are rescaled to unit-length, and the next training example is processed. The algorithm cycles through the training set until the error becomes negligible or no further improvement is observed over a long period of time.

# 4 A Sample Grammar

Preliminary results show that the learning procedure sketched above can indeed learn grammars for both artificial languages and natural language fragments of moderate complexity. As emphasized in the introduction, the results of the learning process can then be analyzed in terms of the context-free formalism VSGs are based on.

As an example consider a fragment of English consisting of transitive sentences ('A circle touches a square') and copula sentences ('A circle is below a square') involving the nouns *circle, square*, the verbs *is, touches*, the prepositions *above, below* and the determiner *a* (this fragment is borrowed from the $L_0$ project domain (Feldman et al., 1990), a sample grammar for it is given below).

The algorithm was run over a set of 6 positive and 18 negative samples, listed in figure 2. the number of rules was set to 5 and the category dimension to 15. At a constant learning rate of 0.5 the error was typically negligible after 50 passes over the training set.

As a method for analyzing the resulting VSG we used *cluster analysis*, which groups vectors according to a distance metric in a hierarchical fashion. Figure 3 shows the result of clustering all vectors occurring in rules as well as the fixed S vector. The graph shows that the vectors fall into nine major clusters of left-hand side and right-hand side rule vectors. Further analysis of these clusters shows not only that they form a rule system that accounts precisely for the input sample, but also that these rules and categories can be put into a one-to-one correspondence with a natural standard CFG for the language at hand:

$$\begin{aligned} S &\rightarrow NP\ VP \\ NP &\rightarrow Det\ N \end{aligned}$$

```
+ ((a circle) (touches (a square)))
+ ((a square) (touches (a circle)))
+ ((a circle) (is (below (a square))))
+ ((a square) (is (below (a circle))))
+ ((a circle) (is (above (a square))))
+ ((a square) (is (above (a circle))))
- (a square)
- (a circle)
- (above (a circle))
- (below (a square))
- (touches (a circle))
- (touches (a square))
- (is (above (a square)))
- (is (above (a square)))
- (is (below (a circle)))
- ((a circle) (below (a square)))
- ((a square) (above (a circle)))
- ((a circle) (is (touches (a square))))
- ((is circle) (touches (a square)))
- ((a circle) (a (a square)))
- ((a square) (is (below (is circle))))
- ((a square) (touches (below (a circle))))
- ((a circle) (is (a square)))
- ((a square) (a (above (a circle))))
```

Figure 2: Training set used for the VSG learning experiment. The data is drawn from a fragment of English generated by the grammar given in the text. Positive training instances are labeled with +, negatives ones with -.

$$\begin{aligned} VP &\rightarrow VT\ NP \\ VP &\rightarrow VC\ PP \\ PP &\rightarrow P\ NP \\ N &\rightarrow square|circle \\ VT &\rightarrow touches \\ VC &\rightarrow is \\ P &\rightarrow above|below \\ Det &\rightarrow a \end{aligned}$$

(Figure 3 explains how CFG symbols map onto vector clusters.)

Of course the details of the resulting rule and category structure are highly dependent on the training environment. For this example, extreme conditions were intentionally chosen to generate the perfect correspondence between the structure learned and the traditional CFG. Specifically, constraining the number of rules to five forces a parsimonious use of categories. With more rules to work with either redundancies would develope (several rules serving the same function) or some rules stay useless (never winning a competition and not converging onto meaningful categories). Also, the relatively large number of negative examples ensured that the categories formed were sufficiently discriminatory. With less or no negative examples a grammar develops that accounts for all the positive examples but fails to exclude all the negative ones, due to overly general rules.

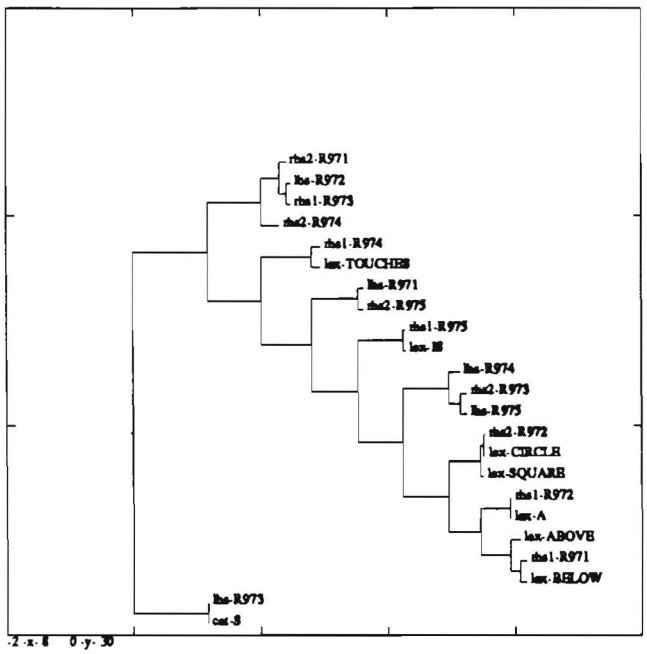



Figure 3: Clusters of category vectors derived from sample language. The rules are arbitrarily numbered R971 through R975, left-hand side vectors are labeled 'lhs', first and second right-hand sides 'rhs1' and 'rhs2', preterminals 'lex'. The nine major clusters correspond (from top to bottom) to the nonterminals NP, VT, PP, VC, VP, N, Det, P, and S.

# 5 Conclusions

These and other examples show that the algorithm sketched above is effective in extracting categories under the tight constraints imposed by the theoretical framework (context-freeness). More importantly, Vector Space Grammars show that traditional theoretical concepts can be generalized to profit from some of the powerful techniques developed in research on connectionist learning. Our ongoing work is geared towards both exploring the possibilities of Vector Space grammatical representations and finding other areas were traditional theories and connectionist methods can 'learn' from each other.

# References

Elman, J. L. (1988). Finding structure in time. CRL Technical Report 8801, Center for Research in Language, University of California at San Diego, La Jolla, Calif.

Elman, J. L. (1989). Representation and structure in connectionist models. CRL Technical Report 8903, Center for Research in Language, University of California at San Diego, La Jolla, Calif.

Feldman, J. A., Lakoff, G., Stolcke, A., and Weber, S. H. (1990). Miniature language acquisition: A touchstone for cognitive science. Technical Report TR-90-009, International Computer Science Institute, Berkeley, Calif. Also appeared in the Proceedings of the 12th Annual Conference of the Cognitive Science Society, pp. 686–693.

Morgan, J. L. (1986). *From Simple Input to Complex Grammar.* MIT Press, Cambridge, Mass.

Morgan, J. L., Meier, R. P., and Newport, E. L. (1987). Structural packaging in the input to language learning: Contributions of prosodic and morphological marking of phrases to the acquisition of language. *Cognitive Psychology*, 19:498–550.

Morgan, J. L., Meier, R. P., and Newport, E. L. (1989). Facilitating the acquisition of syntax with cross-sentential cues to phrase structure. *Journal of Memory and Language*, 28:360–374.

Pollack, J. B. (1988). Recursive auto-associative memory: Devising compositional distributed representations. Technical Report MCCS-88-124, Computing Research Laboratory, New Mexico State University, Las Cruces, New Mexico.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986a). Learning internal representations by error propagation. In (Rumelhart et al., 1986b), pages 318–362.

Rumelhart, D. E., McClelland, J. L., and The PDP Research Group (1986b). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1: *Foundations*. Bradford Books (MIT Press), Cambridge, Mass.

Rumelhart, D. E. and Zipser, D. (1985). Feature discovery by competitive learning. *Cognitive Science*, 9:75–112. Reprinted in (Rumelhart et al., 1986b), pp. 151–193.

Stolcke, A. (in preparation). Vector space grammars and grammatical category acquisition. Technical report, International Computer Science Institute, Berkeley, Calif.

Zadeh, L. A. (1972). Fuzzy languages and their relation to human and machine intelligence. In *Proceedings of the Conference on Man and Computer, Bordeaux, France, June 1970*, pages 130–165. S. Karger, Basel.

# Appendix: Error backpropagation through grammar rules

This section shows how the equations for error backpropagation through the parse tree are derived.

As noted earlier, we can define the result $\mathbf{a}$ of applying a rule $\mathbf{x} \rightarrow \mathbf{y}\ \mathbf{z}$ to two categories $\mathbf{b}$ and $\mathbf{c}$ as

$$\mathbf{a} = (\mathbf{b} \cdot \mathbf{y})(\mathbf{c} \cdot \mathbf{z})\mathbf{x}$$

Consider a general node in the parse tree with category vector $\mathbf{a}$, derived from two child nodes with vectors $\mathbf{b}$ and

c using the rules $\mathbf{x} \rightarrow \mathbf{y}\ \mathbf{z}$. We will consider the general 'activation rule'

$$\mathbf{a} = f(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) \mathbf{x} \tag{4}$$

where $f$ is a differentiable and monotonic 'activation function' as typically used in other connectionist approaches. In the experiments reported $f$ is either linear (the identity) or sigmoid.

An error function $E$ is defined on the root category of the parse tree. If the training sample was a positive one, this is typically the sine of the angle between the the $S$ vector and the root, on negative samples the cosine of that angle. Backpropagation starts by computing $\frac{\partial E}{\partial \mathbf{a}}$ for the root category $\mathbf{a}$ directly from that error function. We use $\frac{\partial E}{\partial \mathbf{a}}$ here as a convenient shorthand for the vector of partial derivatives with respect to the individual components of $\mathbf{a}$,

$$\begin{pmatrix} \frac{\partial E}{\partial a_1} \\ \frac{\partial E}{\partial a_2} \\ \vdots \\ \frac{\partial E}{\partial a_n} \end{pmatrix}.$$

The inductive assumption in the procedure is then that $\frac{\partial E}{\partial \mathbf{a}}$ is already computed, and that the remaining derivatives, $\frac{\partial E}{\partial \mathbf{b}}$, $\frac{\partial E}{\partial \mathbf{c}}$, $\frac{\partial E}{\partial \mathbf{x}}$, $\frac{\partial E}{\partial \mathbf{y}}$, and $\frac{\partial E}{\partial \mathbf{z}}$ can be derived from there. Eventually only the latter three derivative vectors are needed to determine the rule adjustment, but the remaining ones are required as intermediate values in the recursion. The recursion bottoms out at the lexical nodes, since lexical rules have only a left-hand side vector, but no adjustable right-hand side.

From the expanded version of eq. (4),

$$a_j = f(\sum_i b_i y_i) f(\sum_i c_i z_i) x_i, \tag{5}$$

we get

$$\begin{aligned} \frac{\partial E}{\partial b_i} &= \sum_j \frac{\partial E}{\partial a_j} \frac{\partial a_j}{\partial b_i} \\ &= \sum_j \frac{\partial E}{\partial a_j} f'(\mathbf{b} \cdot \mathbf{y}) y_i f(\mathbf{c} \cdot \mathbf{z}) x_j \\ &= \left( \sum_j \frac{\partial E}{\partial a_j} x_j \right) f'(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) y_i \\ &= (\frac{\partial E}{\partial \mathbf{a}} \cdot \mathbf{x}) f'(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) y_i \end{aligned}$$

or, using the shorthand,

$$\frac{\partial E}{\partial \mathbf{b}} = (\frac{\partial E}{\partial \mathbf{a}} \cdot \mathbf{x}) f'(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) \mathbf{y} \tag{6}$$

The situation for $\frac{\partial E}{\partial \mathbf{c}}$ is symmetrical and we get

$$\frac{\partial E}{\partial \mathbf{c}} = (\frac{\partial E}{\partial \mathbf{a}} \cdot \mathbf{x}) f(\mathbf{b} \cdot \mathbf{y}) f'(\mathbf{c} \cdot \mathbf{z}) \mathbf{z} \tag{7}$$

We can now compute derivatives for the components of the rule. For the left-hand side we get

$$\begin{aligned} \frac{\partial E}{\partial x_i} &= \frac{\partial E}{\partial a_i} \frac{\partial a_i}{\partial x_i} \\ &= \frac{\partial E}{\partial a_i} f(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) \end{aligned}$$

or, in shorthand,

$$\frac{\partial E}{\partial \mathbf{x}} = f(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) \frac{\partial E}{\partial \mathbf{a}} \tag{8}$$

The equations for derivatives with respect to y and z are symmetrical to the ones with repsect to b and c, respectively (eqs. (6) and (7)).

$$\frac{\partial E}{\partial \mathbf{y}} = (\frac{\partial E}{\partial \mathbf{a}} \cdot \mathbf{x}) f'(\mathbf{b} \cdot \mathbf{y}) f(\mathbf{c} \cdot \mathbf{z}) \mathbf{b} \tag{9}$$

$$\frac{\partial E}{\partial \mathbf{z}} = (\frac{\partial E}{\partial \mathbf{a}} \cdot \mathbf{x}) f(\mathbf{b} \cdot \mathbf{y}) f'(\mathbf{c} \cdot \mathbf{z}) \mathbf{c} \tag{10}$$

# Knowledge and Language

J. van der Leeuw[1]

Computational Linguistics
Faculty of Arts
University of Amsterdam

## ABSTRACT

Natural language and knowledge are tightly connected. In this paper some fundamental questions concerning the relation between language and knowledge are posed. Such questions are relevant to any research in the field of computational linguistics, including the field of machine learning of natural language. However, most answers are not available, or lead to further questions. But, one thing is certain: language cannot do without knowledge.

Further research into the relation knowledge-language seems unavoidable. The outline of a model is introduced, which forms a foundation for further research. In the model, a multi-layered knowledge base interfaces with the world through a dynamic, context-dependent conceptual model. A meta-cognitive level interprets the interaction with the world on the basis of the conceptual model. It also decides what knowledge to project from the knowledge base into the conceptual model, dependent on the current context. The conceptual model feeds back into the knowledge base. In this way knowledge on higher-order entities (concepts) can be included in the base, and a multi-layered knowledge base emerges.

It might be necessary to develop new logical systems for inferring on the knowledge and the conceptual model. The strict logic of existing systems seems to conflict with the flexibility of the human mind. Reformulation of logical concepts like "inconsistency" seems unavoidable. In the context of language, the term incompatible seems more appropriate. Such incompatibilites in the conceptual model, which is a partial projection of the knowledge base, should be resolved. A non-classical inference mechanism executed by a meta-cognitive level resolves this incompatibility.

The ultimate goal of this research is a data-driven knowledge model which, when implemented, will show gradual "spontaneous" development of language performance. Emphasis is on knowledge. It is therefore likely that other data (e.g. visual) will have to be considered, apart from linguistic data.

The ideas sketched in this paper are still in a state of genesis. They are presented here to provoke discussion. Many aspects of the relation knowledge-language remain undiscussed, e.g. the exact location of language rules in the model.

## BACKGROUND

The author has recently begun the research outlined as a graduate Ph.D student at the university at the University of Amsterdam. He was educated in electrical engineering and theoretical computer science at the Technical University of Delft. During his studies, emphasis was on computational linguistics, with a master's thesis on discourse handling.

The research is carried out at the department of Computational Linguistics, in a project group concerned with data-oriented parsing. Apart from this main research item, research is carried out in the field of connectionism (**Scholtes 91**), gestalt perception, language acquisition (by the author), and efficient implementation techniques.

The department of Computational Linguistics participates in the Institute for Language, Logic, and Information. Among the other participants are the departments of Computer Science and Philosophy of Language. The author maintains relations with the department of Philosophy of Language.

1 Jeroen van der Leeuw
Computational linguistics
Faculty of Arts
University of Amsterdam
Spuistraat 134
1012 VB Amsterdam
The Netherlands

NEZA@ALF.LET.UVA.NL

## INTRODUCTION

Knowledge plays an important role in language performance. Numerous facts show us that language performance cannot exist without world knowledge. As an example, consider the fictitious headline: "Bush stands firm in storm". No one who has read any newspapers lately, will interpret this sentence as: "Scrub survives the windy forces of nature". Interpretation of such sentences draws heavily upon our world knowledge (in this case "world" in the most literal sense).

Such considerations show us **that** knowledge is essential to natural language. However, the main question remains unanswered: **how** is knowledge involved in natural language? What is the relation between our knowledge and natural language performance? Do we translate language utterances into some internal (e.g. propositional) format, and then check the consistency of the translated utterance against our knowledge base? This seems highly unlikely. In the case of the example above, we directly come to the correct reading, instead of testing the possible interpretations[2] against our knowledge base. The context (a headline during the crisis in the gulf) forces the intended reading.

The context prepares us for the correct interpretation. Thus, language analysis seems to be guided by knowledge, and not checked against knowledge. We isolate parts of our knowledge into an expectancy model. When reading the word "Bush" we already know that the president is meant. We would be taken by surprise if the article continued as: "Last Wednesday, a scrubwood was not taken down by heavy wind in the southeast of Texas".

In this paper a model of knowledge is described which incorporates such context-dependent concepts. The described model will serve as the basis for research on the relation between knowledge acquisition and language acquisition. Main goal is to develop a model which, when implemented, will show "spontaneous" language development.

## KNOWLEDGE AND LANGUAGE

Analysis of the interaction between knowledge and language leads to other questions. Can we speak of two diffent systems, a knowledge base and a language faculty, both with specific properties? A number of alternatives exists. When we process some language utterance, it seems psychologically plausible to presuppose some speech recognition component which translates sounds into some other representation (language of thought?). Recent results in speech recognition with the use of connectionist models seem to support this. But what is the nature of this preprocessing? Does it concern syntactic, semantic or even pragmatic analysis as well as phonetic analysis? If so, the argument above implies that such preprocessing would have to be controlled by our knowledge. Can we directly include the output of such a preprocessor in our knowledge base, if such a base exists?

If a preprocessing component does not perform this extensive analysis of utterances, would that not imply the existence of some "language of thought" with semantic - or even syntactic - properties? What should such a language of thought look like?

The nature of our knowledge base seems to be a deciding factor in this analysis. Apart from the question whether we can speak of a separate storage place for knowledge, the question arises how knowledge would be stored in such a system. Is it stored in the form of language-like expressions, (propositions or "language of thought"), in the form of images, in a distributed form in a neural net, or in some other, stil unknown, representation? The connectionist answer is a tempting one. Psychologically, it can be quite succesfully defended. It sheds light on processes like association, memory retrieval, memory loss, the often surprising absence of severe effects of minor brain damage and many more aspects of knowledge.

It seems less plausible that our knowledge is represented by language-like expressions. For these representations can be directly translated into language utterances. But it is common knowledge how difficult it can be to express our knowledge in a precise way. Secondly, one hardly ever uses the same words to express the same knowledge.

It is plausible that knowledge is represented in a distributed system. Does this imply the existence of an independent, separate knowledge system? If we consider language, some interface between language-like expressions and the knowledge system exists. It seems reasonable to assume that the translation takes place in the mind, before expressing the resulting utterance. This would imply that the language faculty and the knowledge base are two distinct systems.

If we consider language acquisition, the list of dreadful questions continues. Which underlying principles of language are innate? Why the discrepancy between the active and the passive language performance of children? Is this just a matter of filtering out the understandable parts of an utterance?

A common conception of children's language acquisition is that child language travels through successive stages of complexity: one-word sentences, two-word sentences, subject-verb-object sentences etc., before reaching adult complexity. However, many children (the author is one of them) show a different kind of development. They start to use language at an advanced age, first talking double Dutch for a short period of time, but in sentences with adult intonation. After this stage, they talk in understandable, relatively complex sentences. Why this remarkable difference in development? It is too simple an answer to assume that the second group silently travels the same stages as the first group, for why should they keep silent? Shyness?

What is the relation between knowledge acquisition and language acquisition? Can we speak of a recurrent process: language acquisition bringing forth knowledge acquisition and knowledge acquisition bringing forth language acquisition? But we should not underestimate the influence of other perceptional stimuli like vision, hearing, smell, taste, touch etc. Does language play the same role as such stimuli or is there a fundamental difference?

How is knowledge acquired and how does its acquisition influence language? Knowledge seems again a crucial factor. But also again a nasty one, for questions rise which cannot (yet) be answered.

The ideas sketched in the rest of this paper have their origin in my dissatisfaction with the current classical language paradigm. The classical approach consists of parsing a sentence (syntax), interpretating the result (semantics), and applying the interpretation (pragmatics). The Holy Trinity of syntax, semantics and pragmatics is a convenient one for language analysis, but seems to be an artificial one for describing natural language understanding. It has been used to strictly separate the three properties of language. I think that a reasonable model should integrate these properties.

The classical approach is outdated. It implies that semantics is influenced by syntax, and pragmatics by semantics and syntax. But, semantic and pragmatic considerations control syntactic analysis, e.g. when resolving ambiguity. Pragmatic considerations affect the semantic analysis of a senten-

2 There are at least twelve possible interpretations. "Bush" can be interpreted as the president, as scrubwood, or as jungle. Both "stands firm" and "storm" can be interpreted in a literal sense, or in a metaphorical sense.

ce: interpretation does not solely depend on structure, but a great deal on context as well. We have seen that knowledge guides language analysis. But where does it fit in?

Language understanding is a complex process, consisting of interwoven processes of feature analysis (concerning syntax, semantics, pragmatics, and knowledge). This situation becomes even more complex in the field of language acquisition.

Until now I have only posed questions without providing the answers. And I doubt it that anyone will be able to answer these questions in the near future. However, these questions are relevant to the field of computational natural language analysis. Insight in human language performance can provide us with strong clues on how to implement natural language systems. Computers cannot compete with humans in activities like language. The reason for this does not lie in storage problems, processor speed, but in more fundamental problems like representation and inference.

As an attempt to bypass the problems discussed, the presented model will be primarily concerned with knowledge. It forms an attempt to tackle the problems at the bottom (knowledge), as an alternative to the top-down syntax-semantics-pragmatics approach. Its basic assumption is that the state of knowledge reflects the state of language and vice versa.

The model presented in this paper is essentially a computationial one. The main goal of this research is to develop a language learning system, which is knowledge-based. However, we should not forget to keep an eye on human language performance for valuable clues on how to (or rather: on how not to) develop computational models.

The ultimate goal of this research project is a knowledge model which, when implemented, will show gradual "spontaneous" development of language performance, whilst explaining some properties of human language acquisition, a most ambitious goal indeed! Such a model would have to cope with knowledge acquisition, knowledge processing, and acquisition of linguistic rules. The model should be formalized such that implementation is easy, and draw upon knowledge about human language acquisition.

## A MODEL OF KNOWLEDGE AND LANGUAGE

I have stressed the importance of knowledge in language related activities. This simple statement brings forth utterly complex questions, which cannot (yet) be answered. One is forced to make choices which might be psychologically implausible, in order to "get something to work".

In this section I will confront the reader with some premature ideas still in the state of genesis, which will hopefully lead to such an inaccurate, but working model. I present them to provoke discussion, not to present research results.

In the model I will postulate a separate knowledge base. As has been argued in the previous section, it is plausible that our knowledge is represented in a distributed form. I will also postulate a conceptual system. This system forms the interface between language and knowledge. In this conceptual system, a language-like representation is assumed. I think that this is the main feature of the presented model: a projection of distributed knowledge into a language-like representation.

What are the properties of this conceptual system? Depending on the context, distributed knowledge is projected into the system. On the level of knowledge, association forms the inference mechanism. On the level of the conceptual system, other inference mechanisms like deduction take place. At this level, inference is conscious. We manipulate symbols instead of having them "pop up".

Consider for instance a game of chess. If we analyze a position, we unconsciously associate and a move "pops up". We then consciously check whether such a move would be a good one. We "see" combinations, and afterwards analyze them for correctness. Such active symbol manipulations are relatively hard. Try to verify checkmate in eight moves!

The crucial implication of the above considerations is that knowledge is on the fly projected into a context-dependent, dynamic system. Abstraction becomes a parametrized process instead of a logical property. In slightly different contexts, our conceptual system might be quite different.

In the conceptual system, we can use the more classical logical approaches to natural language processing, for it is assumed that we use a language-like representation in this system.

Some important parts of the model are still missing. For how to control the association in the knowledge base? And how is the link between language and the conceptual system established? I assume that a meta-cognitive level exists, which controls the association process in the knowledge base, dependent on the current state of the conceptual system and incoming information from interaction with the world. Incoming information is processed by the meta-cognitive level, taking the conceptual system as data. The meta-cognitive level feeds such processed information into the knowledge base. Finally, the meta-cognitive level generates output to the world.

The model can be visualized as:

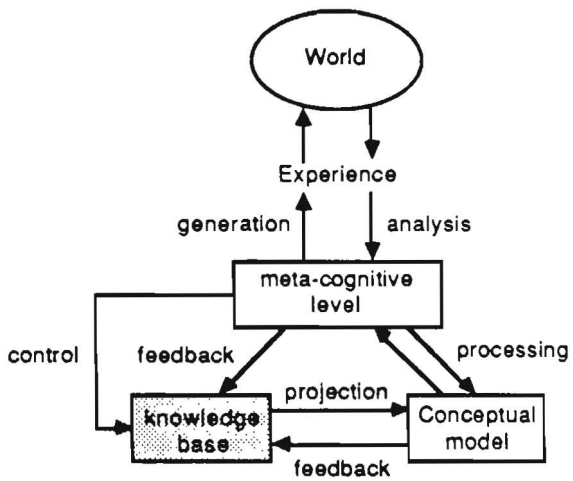



*Fig. 1: Model of the relation knowledge-world.*

The knowledge base is projected into a conceptual model. This projection process is controlled by the meta-cognitive level, such that association takes place in a context-dependent way. Knowledge relevant to the current context is included in the conceptual system in a language-like representation, irrelevant knowledge is not. The projection also involves the translation from distributed knowledge into language-like expressions. It might be necessary to process the chosen knowledge, for instance by providing extra relations between pieces of knowledge or by adding information about certain knowledge. The meta-cognitive level implicitly controls what knowledge to include, change or remove and what information to add by controlling the association process in the knowledge base.

The interaction with the world (experience) is evaluated by the meta-cognitive level on the basis of the conceptual model. The meta-cognitive level takes the conceptual model as data and reasons on the basis of this data in order to analyze input or to generate output. In case of conceptual "crashes"

(e.g. when the headline is followed by the unexpected continuation), the meta-cognitive level will have to decide on what actions to undertake. Thus, complex reasoning will be carried out by the meta-cognitive level, although it seems sensible to ascribe some reasoning capabilities to the conceptual system.

The conceptual model is dynamic, for it is context-dependent. The conceptual system changes when context changes, information will be included, discarded, or play a different role in the conceptual model. The projected conceptual system is relevant to a certain situation. If a situation changes, its related model changes as well.

New experiences can be analyzed by the meta-cognitive level and the conceptual system and then be included in the knowledge base. "Spontaneous" processing of the knowledge base (meta-cognitive controlled self-organization) might also lead to changes in the system. Structures in the knowledge base can emerge, relations found etc., by this controlled self-organization.

The meta-cognitive level and the conceptual model feed back into the knowledge base. Thus, knowledge can be added or changed. A second effect of this feedback is that it ensures a multi-layered knowledge base, for knowledge about concepts in the conceptual system can be fed into the base. Higher-order entities in the conceptual system can play the role of basic entities in the base.

In this research, language will be emphasized as the means for interaction with the world, both passive and active. However, it is not clear whether such a restriction will lead to a reasonable model. Visual contact with the world, for instance, seems to be an important factor in knowledge acquisition and conceptual development. It might therefore be unavoidable to incorporate emulation of visual interaction in the model as well.

The question remains where and how language exactly interacts with the conceptual system. I cannot provide an answer yet. Are linguistic rules included in the meta-cognitive level such that utterances are translated into a "language of thought"? Do we check the resulting language-of-thought expressions against the conceptual system? At this moment this approach seems a sensible one. The acquisition process is then implemented by evolution of the knowledge base and the meta-cognitive level (both affecting the conceptual model).

## HUMAN KNOWLEDGE BASES

The conceptual system has a language-like representation and is governed by inference processes like deduction. But, if we view such bases as logical systems, we encounter severe problems with the rigid notions of logic. Consider for instance the notion of inconsistency. If a set of logical expressions is inconsistent, in classical logic one can deduce anything from it (ex falso sequitur quodlibet) and in intuitionistic logic one can deduce nothing from it.

This problem emerges when using propositional knowledge representation. Distributed and imagery representation do not suffer from this problem. The problem lies in the logical system, not in the human mind. The research will try to reformulate logical notions such that better correspondence exists between the used logical systems and the model of human knowledge processing. As an example, in the case of inconsistency, it seems better to speak of incompatibility between diffent layers (levels).

Consider for instance a knowledge base which contains the information that birds fly, that an ostrich is a bird and that ostriches do not fly. When asked whether birds fly, the best answer seems to be: "Yes". This conforms to the Gricean maxims, for with this answer we cover the greater part of the bird population. We access the knowledge base on the level of the bird class. If the question follows: "But how about ostriches?", we answer: "No, they don't, but they are an exception". We are forced to shift our attention towards a different level, that of the primitive class. At this level we access the specific information about ostrichs to conclude that they form an exception to the general rule accessed before.

Incompatibility between the level of birds and that of the primitive class exists. When incompatibilty is involved (in the case of the second question), we can resolve this incompatibility by reasoning that ostriches form an exception (at the meta-cognitive level). With such an approach to knowledge more abstract and declarative descriptions of phenomena like prototype theory, default logic, and concept formation theories like in **Bartsch (90)** follow implicitly from the model in a more procedural way, thus explaining why instead of stating that it is the case.

As may be concluded from the considerations above, the knowledge base might not conform to the classical logical form. Alternative knowledge representations and non-monotonic reasoning might prove to be more useful, e.g.:

Classical:
$\forall x$ (ostrich(x) $\rightarrow$ bird(x))
$\forall x$ (ostrich(x) $\rightarrow \neg$ fly(x))
$\forall x$ (bird(x) $\rightarrow$ fly(x))
$\forall x$ (sparrow(x) $\rightarrow$ bird(x))

Alternative:
**bird**(ostrich)
$\neg$ **fly**(ostrich)
**fly**(bird)
**bird**(sparrow)

The classical base is inconsistent. In the alternative case we use predicates over **sorts**. This means that deduction cannot automatically take place. When confronted with a question about ostriches, we first review the data on ostriches. Well, they do not fly and thus deduction is finished. But, if we ask whether sparrows fly, we do not find the data at this level. We are then allowed to relate the predicate bird to the sort bird. In such cases, we can infer the corresponding classical rules from **bird**(sparrow) and **fly**(bird): $\forall x$ bird(x) $\rightarrow$ fly(x) and $\forall x$ sparrow(x) $\rightarrow$ bird(x), and it readily follows that sparrows fly.

Such a deduction system is efficient: special properties of the species (positive and negative facts) are included at the species level, common properties at a higher level. The deduction mechanism described forces us to review the most specific data first: inconsistency is resolved.

The reasoning involved takes place in the meta-cognitive level. Here we decide which lebel to access, what to do when deduction fails, which higher level then to access etc. The facts are included in the conceptual system. They represent our conceptual declarative conceptual knowledge.

Humans are able to reason about conceptual deviations. We would not want to lose this property in our model. The meta-cognitive level should not only be able to control the deduction process, but also to perform meta-logical operations. If one asks whether birds fly, and then whether ostriches fly, the meta-cognitive level will have to realize that an exception is involved.

## CONCLUSIONS

In this paper I have stressed the importance of a knowledge-oriented approach to natural language. In the field of language acquisition this approach seems to be of the utmost importance. I have outlined some proposals for a knowledge model.

An important aspect of the model is the role of the knowledge base in it, and how information in the base is interpreted. I propose a model in which knowledge about entities of different levels of abstraction can be included. Such levels emerge naturally when a feedback from the conceptual system is included in the model.

The approach taken here might have important consequences for the logic involved. It might be necessary to develop a new kind of logical interpretation in order to make the system work. I do not see this as problematic. The ideas expressed in this paper are heavily inspired by years of dissatisfaction with the gap between rigid logic and flexible humanity. The approach is also an effort to bring the two together again.

## REFERENCES

Bartsch, R, 1990, *Concept Formation and Concept Composition*. Report LP-90-03, Institute for Language, Logic and Information, University of Amsterdam.

Scholtes, J.C., 1991, *Learning Simple Semantics by Self-Organization*. Submitted to AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, Stanford California.

# Connectionist semantics for Miniature Language Acquisition

Susan H. Weber
International Computer Science Institute

February 1990

## Abstract

This paper presents an outline of how to build a miniature language learning system, along with some preliminary results on learning selected system components.

The $L_0$ research effort at ICSI focuses on the problem of building a language learning system for the toy domain of simple geometric scenes. Our initial target problem, the Miniature Language Task (MLA), is to learn a fragment of an arbitrary natural language from training examples of descriptive sentences paired with pictures of simple geometric objects. As the focus of this effort is language acquisition and not computer vision, the system starts out with an internal representation for visual scene semantics. Individual scenes are presented in a form the system already understands; its task is to establish the syntax and semantics of the accompanying (partial) linguistic descriptions in the given target natural language. If there are biases in description emphasis characteristic of a particular language, these should be learned along with the grammar syntax and lexeme semantics.

The $L_0$ project at ICSI is undertaken jointly with members of the UC Berkeley Computer Science and Linguistics Departments. The project is headed by Professor Jerome Feldman, the director of ICSI and a professor of Electrical Engineering and Computer Science at UC Berkeley (Feldman et al., 1990a; Feldman et al., 1990b). Adèle Goldberg, a doctoral candidate in Linguistics at UC Berkeley, is gathering cross-linguistic data relevant to the task. Professor George Lakoff of the UC Berkeley Linguistics Department brings to $L_0$ a comprehensive knowledge of all sorts of exotic and obscure world languages. Terry Regier, a doctoral candidate in Computer Science at UC Berkeley, is looking into the acquisition of image-based lexical semantics for closed-class polysemous lexemes describing spatial relations (e.g. prepositions, verbal prefixes) (Regier, 1990; Regier, forthcoming). Andreas Stolcke, a doctoral candidate in Computer Science at UC Berkeley, working on learning syntactic categories (Stolcke, 1990; Stolcke, 1991). Susan Weber, a post-doctoral fellow at ICSI, works on the spatio-temporal semantics of the $L_0$ domain (Weber and Stolcke, 1990; Weber, 1990).

## References

Feldman, J. A., Lakoff, G., Stolcke, A., and Weber, S. H. (1990a). Miniature language acquisition: A touchstone for cognitive science. In *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, pages 686–693. MIT, Cambridge, Mass.

Feldman, J. A., Weber, S. H., and Stolcke, A. (1990b). A testbed for the miniature language $L_0$. In *Proceedings of the 5th Rocky Mountain Conference on Artificial Intelligence*, pages 25–30, New Mexico State University, Las Cruzes, N.M.

Stolcke, A. (1990). Learning feature-based semantics with simple recurrent networks. Technical Report TR-90-015, International Computer Science Institute, Berkeley, Ca.

Stolcke, A. (1991). Vector space grammars and the acquisition of syntactic categories. in the working notes for AAAI-91 Symposium on Connectionist Natural Language Processing, Stanford University, March 26-27, 1991.

Weber, S. H. (1990). Acquiring categorical aspects: a connectionist account of figurative noun semantics. In *Proceedings of the 5th Rocky Mountain Conference on Artificial Intelligence*, pages 295–300, New Mexico State University, Las Cruzes, N.M.

Weber, S. H. and Stolcke, A. (1990). $L_0$: a testbed for miniature language acquisition. Technical Report TR-90-010, International Computer Science Institute.

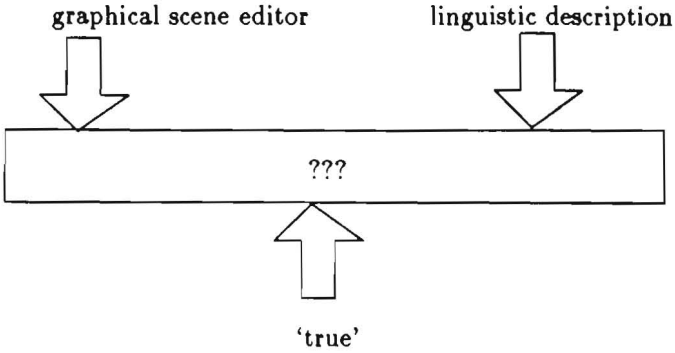
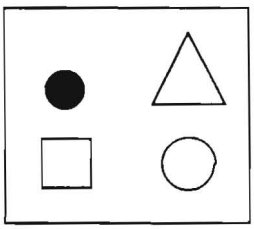



Figure 1: The task is to learn both the syntax and semantics of a natural language fragment from simple scene descriptions with only positive training examples.



A dark circle is above a square.

Un triangle est à droit d'un cercle.

Um cerculo esta embaixo de um triangulo

Figure 2: Training input to the MLA task: a sample picture and several possible partial descriptions, one in English, one in French and one in Portuguese. Training input for a given language would consist of multiple scenes each with assorted descriptions in the target language.

# 1 The $L_0$ testbed

We are investigating the problem of learning a natural language fragment from training examples consisting of simple geometric scenes accompanied by (correspondingly simple) true descriptions in the target language (see Figure 1). An example of system input appears in Figure 2: any given scene can be labelled in a wide variety of ways, depending on the elements being described.

We are assuming that natural languages induce a finite number of decompositions or categorizations of simple visual stimuli. We posit the existence of a set of cognitive primitives that can be combined to produce any known linguistic description of simple geometric scenes. We intend to catalogue these primitives by analyzing the overlap between concepts acquired for a reasonably large set of natural languages. If our hypothesis is correct, this cataloging process will soon converge on the common underlying representations.

Our initial target language is a fragment of English known as $L_0$. A non-learning prototype system has been implemented in Prolog as a testbed for the components of the eventual learning system. The interfaces between the language, vision and semantic representation components will crucially determine the ease with which language can be acquired.

The system's architecture is sketched in Figure 3. Two of the three input sources of the hypothetical learning system are still in use, but the testbed, modeling a fully trained learner, produces yes/no truth assessments of the linguistic input (the 'description' is treated as a query). The internal structure of the testbed provides us with a working blueprint for the structure of the target learning system. Components include: categorical feature vectors, object representations generated by an interactive scene design session on the graphics interface;

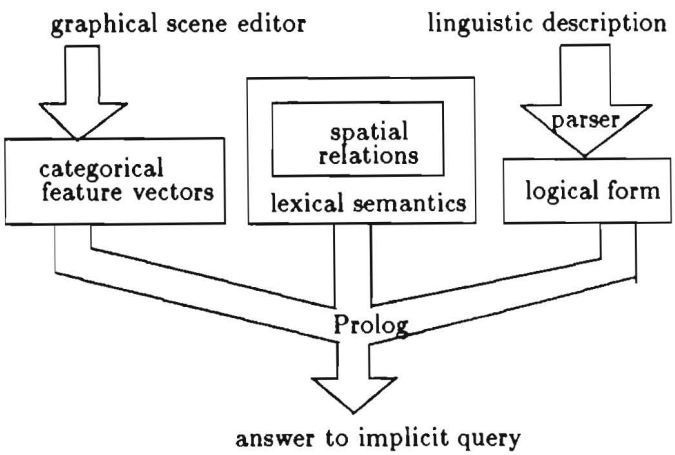



Figure 3: Representational fixpoints chosen to facilitate learning the inter-representational mappings.

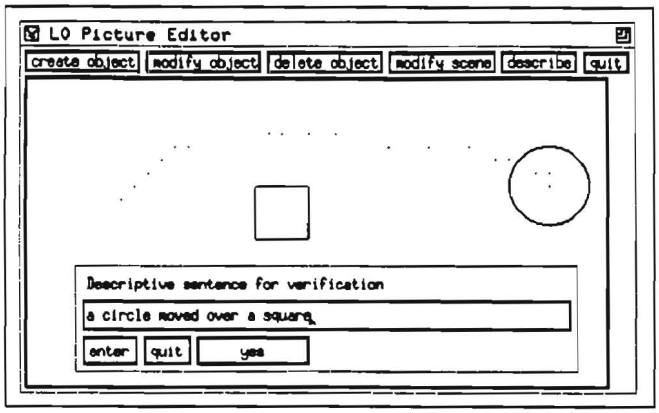



Figure 4: The graphical scene editor with an English language query.

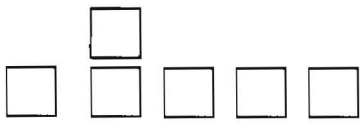


Figure 5: Object reference by relational properties: the referent of "the square on top of another" should be available in constant time; however, in the Prolog testbed the time to resolve the reference is quadratic in the number of objects in the field.

spatial relations, eg. how the region 'above' a landmark object is defined; parsing into a logical form, i.e. the assignment of lexemes to an appropriate syntactic category; and lexical semantics, the association of each syntactic component with its corresponding spatial relation or categorical feature value.

A graphical interface (see Figure 4) allows a user to draw scenes involving circles, squares and triangles, and to pop up a window for the natural language 'query' on the given picture. Since object features can be modified, both present and past tense queries are supported, as are static (eg. *is above*) and dynamic (eg. *moved onto*) relations. From the system's standpoint, the 'visual' input is a collection of facts about particular feature value assignments, eg. at time step 2 there is an object at location (10, 30) with circular shape, light shade and radius of 5. The linguistic description is parsed and translated into an internal logical form which is representationally compatible with the scene data. Once the mapping between the internal linguistic representation and the internal visual representation has been established, the task is complete. The question is how to achieve the desired mapping.

The crux of the mapping problem is the relational nature of natural languages. Virtually all naturally descriptive predicates in a simple spatial domain are relational in nature. Even at the level of linguistic reference, it turns out that while objects can be physically pointed to for diectic reference, purely verbal forms of communication must rely on indirect methods of establishing reference identity. When a unique property value (or conjunction of values) exists in the frame of reference, that value can be named, eg. the large light square. When property values alone leave ambiguity of reference, however, relational properties must be resorted to, eg. the square below the circle. In the visual domain, however, information is available in terms of scene geometry and categorical feature values eg. position and radius. The question is then how to transform this categorical information into the relational form used in language?

There are two approaches to solving this matching problem. The one used in the Prolog testbed is to dynamically determine which visual relation is being referred to. This involves searching the object feature vector space until an appropriate set of vectors is found which satisfies the relational definitions referred to in the query. The flaw with the approach is it does not correctly handle reference by relational property (see Figure 5). The second option is to tabulate all the relational information in the scene, then perform unification with the linguistic input. This approach can turn out to be combinatoric unless attentional selectivity in the linguistic input is exploited to focus and direct the visual processing.

The testbed system is being used as a platform to test out learning components. There are three components under development, a static model of lexical semantics, learning grammar syntax and learning spatial relations.

## 2 Lexical semantics

An alternative to relying on object reference to index and verify relations is to have the capacity to tabulate all the relations in the picture and use the linguistic input to edit out the irrelevant ones. In this scenario, combinatorics are avoided by having a fixed sized buffer and relying on the linguistically supplied focus of attention (much as the eye would rely on foveating) to load only currently relevant information into the buffer.

This situation suggests an obvious bootstrapping strategy: before the linguistic knowledge exists to offer editorial control (and in the absence of visual focus of attention), the visual scenes must be kept extremely simple, to avoid confusion due to cross talk. That is, noun and adjective semantics must be acquired before tackling relational properties. As the training input grows progressively more complex, the fact remains that unless the linguistic tag contains an unambiguous reference, either to an object or to a relation, the picture semantics is liable to be too noisy for the system to be able to frame

any reasonable learning hypothesis as to the semantics of the unknown lexemes.

The architecture proposed is shown in Figure 6. Visual scene analysis proceeds in three stages. First each potential landmark object in the scene is allocated its own set of primitive relational maps. Relations currently supported are: *touches, near, far,* and the two categorical definitions for each of *above, below, left-of* and *right-of* shown in Figure 7. These maps are defined by differing patterns of connectivity such that for any location of the input stimulus the appropriate region will be generated by spreading activation. Once these regions have been established, the activity in the maps decays to nothing unless an externally excitatory signal inhibits the decay process, where the signal in question is drawn from the set of all potential trajector objects. After this stage the activity in the map reflects the landmark's participation in the given relation. That is, if a landmark's *above* map displays activity, then there is some trajector object above it.

The third stage involves feedback loops to the trajector and landmark controllers, as well as a winner-take-all competition among the map activity summation units. The latter enhances the salience of any unique relations in the scene, while the former, by reducing the number of inputs to the system, may assist in narrowing down the target semantics. The entire process iterates until no further changes are seen at the controller level (trajector, landmark and w.t.a. nodes). At this point it will hopefully be possible for the system to form a reasonable learning hypothesis as to the semantics of the unknown lexeme.

Note that this proposal will result in the system being able to acquire linguistically inspired distinctions between relational descriptions. For example, speakers of one language group may tend to adopt the half-plane definition of 'below', while others may favor the vertical extension definition (see Figure 7).

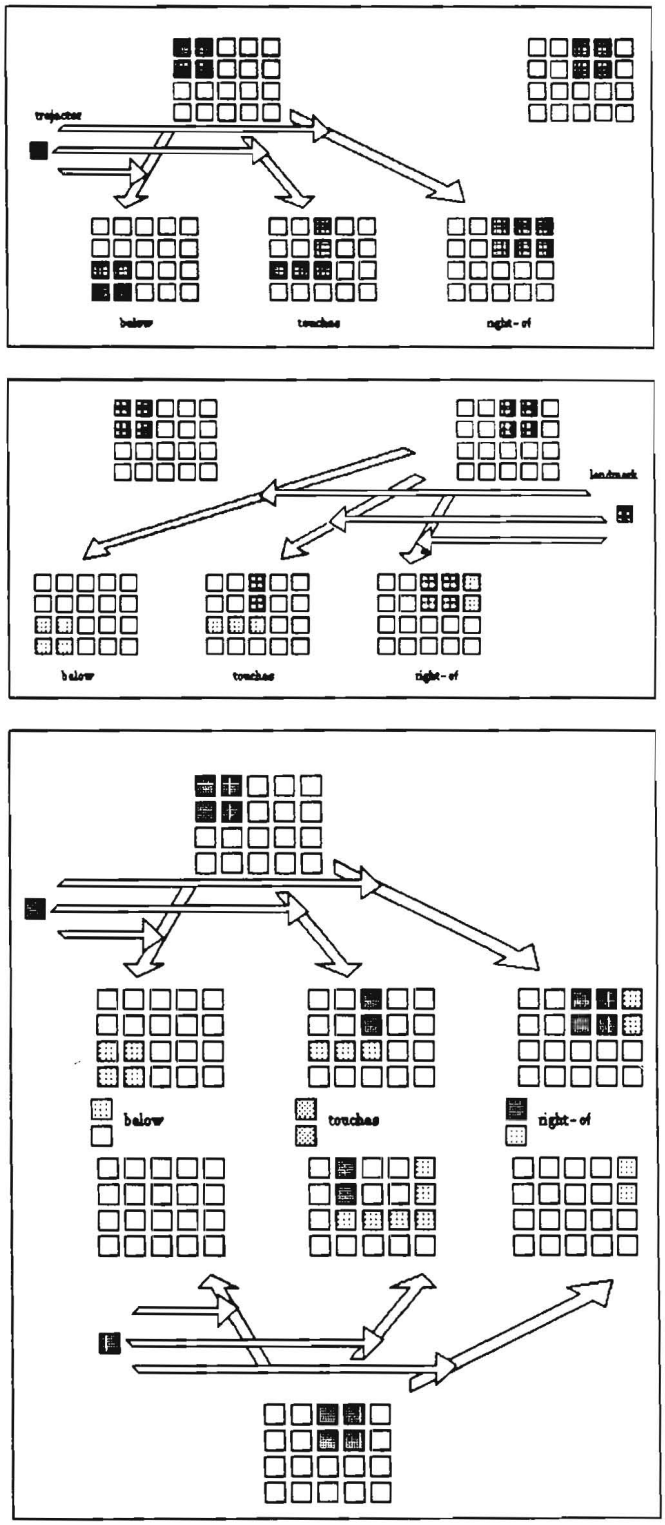



Figure 6: Relational information is tabulated in three stages. First all primitive relations are generated in a dedicated set of maps. Then the maps are edited to reflect overlap with all possible trajector objects. Finally the map activity summation units enter a winner-take-all competition, providing salience to unique relations, and feedback is sent to the trajector and landmark controllers.

# 3 Learning grammar syntax

As a preliminary investigation into the difficult problem of classifying lexemes into lexical categories, Andreas Stolcke ran an experiment with Elman style recurrent nets. The task was to derive a semantic feature vector or slot-filler representation from sequential word level input. There were three slots used: first argument, relation and second argument. Arguments, as object references, have at least one feature value (eg. circle, dark, small) and relations map into known quantities (eg. below, left-of). The system performed well on sentences with tail recursion like "the circle is above the square below the triangle", and even on sentences with one layer of center embedding, like "the circle above a square is touching a triangle", but failed on sentences with center embedding of depth greater than 1, such as "the circle

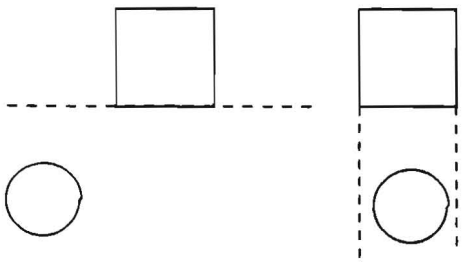


Figure 7: Two possible categorical definitions for 'below'. In the looser version, any object in the half-plane defined by the object's lower surface is 'below' it. In the stricter form, an object must be in the area defined by the vertical downward extension of the landmark reference. A graded form of the concept would presumably combine the two; see Figure 9.

---

above the square touching a triangle is below a circle". The system would incorrectly interpret this to mean "a triangle is below a circle". While humans exhibit increasing difficulty in parsing sentences as the depth of the center embedding grows, this behaviour is unrealistically brittle. This brittleness can be attributed to an inherent deficiency in Elman nets, with their fixed width "sliding window" of attention to the input. For any fixed Elman architecture, a sentence can be constructed whose center embedding is deep enough to sever the connection between the subject and the predicate.

The Elman net approach having proved too limited, Stolcke is currently experimenting with Vector Space Grammars (VSGs), an approach to grammar learning where syntactic categories are represented as points in a continuous metric space. VSGs are a generalization of standard phrase-structure grammars that uses continuous vectors instead of symbols to represent nonterminals in grammar rules. The goal of this generalization is to make the formalism suitable for adaptive learning techniques inspired by connectionism, such as competitive learning and error backpropagation. In contrast to other PDP approaches to this problem, the structure of the grammar is explicitly constrained; for example, the grammar is forced to be context free. Results of using this approach appear promising; details appear in (Stolcke, 1991).

# 4 Learning Spatial Relations

Different languages impose different structurings on physical space. For example, Mixtec is a Mexican Indian languages in which common English spatial concepts such as "above" and "below" are entirely missing. They are replaced by a system of locative terms which does not map at all straightforwardly onto the English system (Brugman, 1983). There are also discrepancies among the spatial systems of closely related languages, such as English, German, and Dutch (Bowerman, 1989). Thus, a significant part of the $L_0$ task is learning the system of spatial concepts embodied in the language being learned.

Terry Regier is developing a connectionist learning system that learns such systems of spatial concepts. The system currently works for single points located relative to some object; the system is being extended to handle full objects located relative to other objects.

The system has so far learned a system of eight English concepts (*above, below, left, right, in, out, off,* and *on*), and several concepts from other languages as well, including Mixtec. Figure 8 presents three of the eight English spatial concepts learned. In this figure, the triangle is to be seen as the reference object (that object with respect to which other objects are located). The size of the black circles indicates the appropriateness, as judged by the system, of using a particular term to describe each position in space.

The system learned these concepts in the absence of explicit negative evidence, as discussed in (Regier, 1990; Regier, 1991). Note also that the system's training set did not include any triangles, and that the system nevertheless correctly generalizes to scenes involving triangles as reference objects.

# 5 Conclusions

The $L_0$ task, originally posed as a 'touchstone' problem for cognitive science, is proving as challenging and rewarding as originally hoped. Our attempt to solve this deceptively simple task has split into three distinct efforts, lexical syntax, lexical semantics and spatial semantics. The obvious next step is to somehow harness the three learning components together as a true test of the soundness of the proposed solution paradigm.

# References


Bowerman, M. (1989). Learning a semantic system: What role do cognitive predispositions play? In et al, M. L. R., editors, *The Teachability of Language*, pages 133–169. Paul H. Brookes, Baltimore.

Brugman, C. (1983). The use of body-part terms as locatives in chalcatongo mixtec. in Report No. 4 of the Survey of California and other Indian Languages, pp. 235-90. University of California, Berkeley.

Feldman, J. A., Lakoff, G., Stolcke, A., and Weber, S. H. (1990a). Miniature language acquisition: A touchstone for cognitive science. In *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, pages 686–693. MIT, Cambridge, Mass.

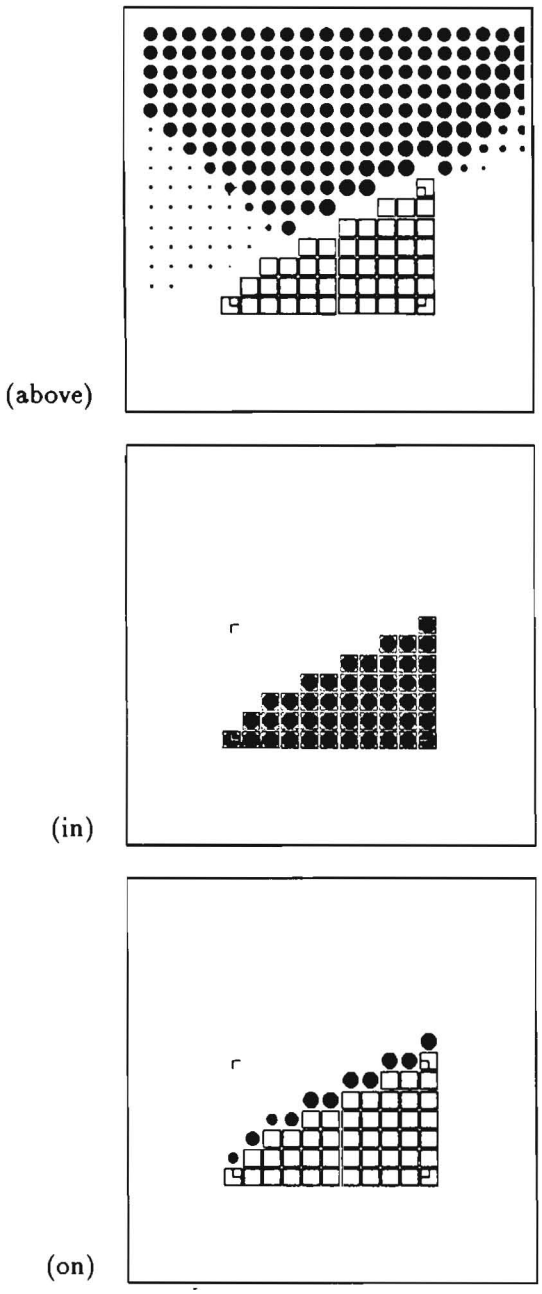



Figure 8: Three English spatial concepts learned by Regier's system

Feldman, J. A., Weber, S. H., and Stolcke, A. (1990b). A testbed for the miniature language $L_0$. In *Proceedings of the 5th Rocky Mountain Conference on Artificial Intelligence*, pages 25–30, New Mexico State University, Las Cruzes, N.M.

Regier, T. The acquisition of lexical semantics for spatial terms. PhD dissertation, Computer Science Division, EECS Dept, University of California at Berkeley, in preparation.

Regier, T. (1990). Learning spatial terms without explicit negative evidence. Technical Report 57, International Computer Science Institute, Berkeley, California.

Regier, T. (1991). Learning perceptually-grounded semantics in the $l_0$ project. In *Proceedings of the 1991 AAAI Spring Symposium on Connectionist Natural Language Processing.*

Stolcke, A. (1990). Learning feature-based semantics with simple recurrent networks. Technical Report TR-90-015, International Computer Science Institute, Berkeley, Ca.

Stolcke, A. (1991). Vector space grammars and the acquisition of syntactic categories. In *Proceedings of the 1991 AAAI Spring Symposium on Connectionist Natural Language Processing.*

Weber, S. H. (1990). Acquiring categorical aspects: a connectionist account of figurative noun semantics. In *Proceedings of the 5th Rocky Mountain Conference on Artificial Intelligence*, pages 295–300, New Mexico State University, Las Cruzes, N.M.

Weber, S. H. and Stolcke, A. (1990). $L_0$: a testbed for miniature language acquisition. Technical Report TR-90-010, International Computer Science Institute.

# Learning and Representing Natural Language Phrases in a Hybrid Symbolic/Connectionist Approach

Stefan Wermter
Department of Computer Science
University of Dortmund
4600 Dortmund 50
Federal Republic of Germany

## General Overview

Our general research interests include the representation of natural language using connectionist and symbolic methods. Our approach aims at evaluating and integrating properties of symbolic and connectionist architectures. Primarily, we concentrate on syntactic and semantic representations focusing on structural disambiguation and semantic classification. As a general task we chose the analysis of phrases. Phrasal analysis often can not rely on as much predictive top-down knowledge as complete sentence analysis and therefore more bottom-up analysis is needed. In this context, connectionist networks appear to be a particularly useful method for learning and representing necessary knowledge for a bottom-up analysis. Using online available corpora and library classifications we designed several hybrid symbolic/connectionist architectures. As examples for *structural disambiguation* we focused on prepositional phrase attachment and coordination using localist relaxation networks, distributed plausibility networks, and a symbolic chart parser. As examples for *semantic classification* we designed a combination of a preprocessing chart parser with a connectionist autoassociator as well as a connectionist architecture using recurrent sequential classification networks. These architectures allow the combination of predefined symbolic knowledge with learned connectionist knowledge for natural language processing.

## Representative List of Publications

Wermter, S. 1989. Integration of Semantic and Syntactic Constraints for Structural Noun Phrase Disambiguation. *International Joint Conference on Artificial Intelligence*, Detroit.

Wermter S., Lehnert W.G. 1989. A Hybrid Symbolic/Connectionist Model for Noun Phrase Understanding. *Connection Science 1,3*. Also available in: Wermter S. 1989. The Analysis of Natural Language Concepts with Connectionist/Symbolic Techniques. Technical Report COINS TR-89-117, University of Massachusetts, Amherst.

Wermter, S. 1989. Learning Semantic Relationships in Compound Nouns with Connectionist Networks. *Proceedings of the Annual Conference of the Cognitive Science Society*, Ann Arbor.

Wermter S. 1990. Combining Symbolic and Connectionist Techniques for Coordination in Natural Language. In: Marburger (Ed.) *Proceedings of the 14th German Workshop on Artificial Intelligence*, Eringerfeld, FRG.

Wermter S. 1991. Learning to Classify Natural Language Titles in a Recurrent Connectionist Model (submitted).

## Abstract

This paper describes a hybrid architecture which uses symbolic and connectionist representations for the structural disambiguation of noun phrases. As a representative example for a whole class of structural attachment problems we focus on coordination (constructions with conjunctions). The architecture combines a symbolic chart parser with connectionist plausibility networks for dealing with coordination. While the symbolic modul supports the sequential compositional syntactic representation, the connectionist modul learns and represents the semantic control knowledge which can modify preliminary syntactic structures. Since other problems like the attachment of prepositional phrases, verb phrases, and relative clauses are very similar, this architecture can be extended for other structural disambiguation problems. The architecture allows for preserving domain-independent syntactic knowledge and learning domain-dependent semantic control knowledge.

# 1 Constraints for Structural Disambiguation

In this section we will focus on structural disambiguation of coordination[1], and we will demonstrate that coordination is just one class of typical attachment problems involving prepositional phrases, verb phrases, and relative clauses. While verbs in sentences can have semantic top-down preferences for subsequent constituents [Wilks 75], noun phrases have less preferences than complete sentences and we have to rely more on semantic bottom-up plausibilities for different coordinations. Consider the following example:

(1) Systems using transistors and transductors

Example (1) contains neither enough syntactic constraints nor verb-related semantic preferences to resolve the coordination. However, the semantic plausibility that "transistors and transductors" are coordinated is higher than the plausibility that "systems and transductors" are coordinated because "transistors and transductors" are similar electric objects while "systems" is a more general term.

The following phrases illustrate that similar structural ambiguities occur in different constructions involving prepositional phrases, verb phrases, and relative clauses. In examples (2) and (3) the prepositional phrase at the end can attach to two different preceding nouns. The same holds for the verb phrase at the end of examples (4) and (5) and for the relative clause at the end of the examples (6) and (7).

(2) Symposiums on hydrodynamics in the ionosphere

(3) Symposiums on hydrodynamics in the auditorium

(4) Symposiums about spacecrafts sent in orbit

(5) Symposiums about spacecrafts held in Germany

(6) Symposiums about spacecrafts which are shot in orbit

(7) Symposiums about spacecrafts which are held in Germany

Since on the one hand such noun phrases show a great deal of sequentiality, compositionality, and recursiveness (symbolic properties) and on the other hand a somewhat restricted complexity for learning graded semantic relationships (connectionist properties), hybrid modeling [Dyer 88] [Hendler 89] [Wermter and Lehnert 89] promises to be a particularly useful approach. In the next section we will see how syntactic and semantic constraints can be implemented in a hybrid model for coordination.

# 2 Syntactic Constraints in a Symbolic Chart Parser

Syntactic constraints determine how a syntactic structure is composed of its parts. Since compositionality and sequentiality are inherent properties of a symbolic representation we implemented a context-free grammar for noun phrases for a symbolic bottom-up chart parser based on [Winograd 83] [Gazdar and Mellish 89]. This chart parser generates a preliminary syntactic structure and deals with simple forms of coordination which can be solved syntactically, for instance coordinated prepositional phrases.

```
PP -->  PP CONJ PP
```

Using this rule the parser builds the following syntactic structure for example (8):

[1] The following description is partly based on [Wermter 90].

(8) Electron collision frequencies in nitrogen and in the lower ionosphere

```
(NP (NG (NN (N ELECTRON)
            (NN (N COLLISION)
                (NN (N FREQUENCIES)))))
    (PP (PP (P IN)
            (NP (NG (NN (N NITROGEN)))))
        (CONJ AND)
        (PP (P IN)
            (NP (NG (DET THE)
                    (ADJG (ADJ LOWER))
                    (NN (N IONOSPHERE)))))))
```

In this syntactic structure we see that the two prepositional phrases "in nitrogen" and "in the lower ionosphere" are coordinated because their syntactic categories PP are at the same level in the preliminary structure above. These examples illustrate that a syntactic chart parser can resolve some simple structure-dependent forms of coordination.

# 3 Semantic Constraints in Connectionist Plausibility Networks

In the absence of clear syntactic constraints and semantic top-down preferences, we rely on the plausibility of semantic coordination relationships. Plausibility networks can learn semantic relationships between two coordinated nouns in a fully-connected architecture shown in figure 1. The input layer consists of 32 input units for two nouns in a coordination relationship and each noun is represented with 16 binary semantic features. We extracted semantic features based on the NASA thesaurus [NASA 85] and developed the following 16 semantic features for noun phrases from the scientific technical NPL corpus [Sparck-Jones and VanRijsbergen 76]: measuring-event, changing-event, scientific-field, property, mechanism, electric-object, physical-object, relation, organization-form, gas, spatial-location, time, energy, material, abstract-representation, empty. The hidden layer consists of 12 units and the output layer has one unit[2]. The real-valued output unit indicates if a coordination relationship between two nouns[3] is plausible (values close to 1) or if it is implausible (values close to 0). For instance, the noun phrase "Systems using transistors and transductors" has the following plausible and implausible coordination relationships.

```
transistors COORDINATED_WITH transductors   1  (plausible)
systems     COORDINATED_WITH transductors   0  (implausible)
```

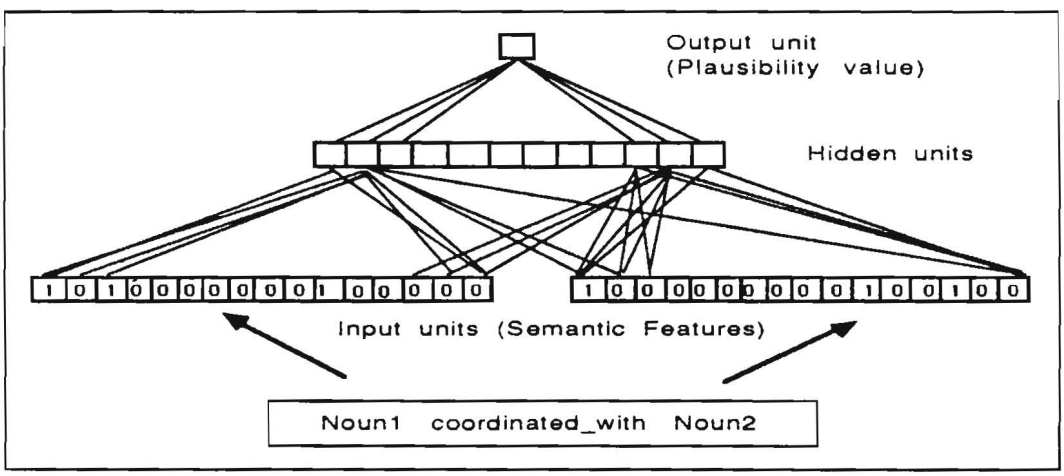



Figure 1: Plausibility Network for Coordination Relationships

[2]Other architectures with 1 to 18 hidden units were tested and the architecture with 12 hidden units performed best.
[3]For compound nouns, only the last noun (the headnoun) is integrated in the coordination relationship.

This plausibility network was trained and tested with coordination relationships of 53 noun phrases from the NPL corpus. There were 40 noun phrases (92 training instances) in the training set and 13 noun phrases (29 test instances) in the test set. The representations of the test instances had not been in the training set. Each training instance consisted of the 32 semantic features for the two nouns and the plausibility value for the coordination relationship. The plausibility value was set to 1 if the coordination relationship was plausible, otherwise it was set to 0.

The network was trained for 800 epochs using the backpropagation learning rule [Rumelhart et al 86] with the learning rate 0.01 and the weight change momentum 0.9. Three different training runs were performed to be more independent from the different start initializations of the network. The average of the total sum squared error over all training instances could be reduced during the learning phase from values of 32.5 to values of 3.2. A training instance was considered correct if the generated plausibility value was higher than 0.5 for a plausible coordination relationship (desired value 1) and lower than 0.5 for an implausible coordination relationship (desired value 0) After 800 epochs the average percentage of correctly learned training instances was 94.2% and the average percentage of correctly classified unknown test instances was 78.2%. In the next section we will describe how this learned knowledge is used for resolving coordination problems.

# 4 Coupling the Constraints

In our hybrid model a chart parser and a plausibility network interact for coordination problems. The chart parser generates a preliminary syntactic structure according to the Right Association strategy [Frazier and Fodor 78] which assumes that a constituent attaches to the directly preceding constituent. In this step, some coordinations are resolved based on syntactic constraints, e.g., coordinations of prepositional phrases as shown in the example above. Right Association is used if no semantic knowledge is available. If more specific semantic coordination relationships exist, they can overrule the Right Association strategy. The following example shows the preliminary syntactic structure together with the plausibilities of the semantic coordination relationships.

```
(9) Fading of satellite transmissions and ionospheric irregularities

Preliminary syntactic structure:
    (NP (NG (NN (N FADING)))
        (PP (P OF)
            (NP (NG (NN (N SATELLITE)
                        (NN (N TRANSMISSIONS))))
                (CONJ AND)
                (NP (NG (ADJG (ADJ IONOSPHERIC))
                        (NN (N IRREGULARITIES)))))))

Semantic Relationships:
    Transmissions COORDINATED_WITH irregularities   (implausible)
    Fading COORDINATED_WITH irregularities          (plausible)

Final syntactic structure:
    (NP (NG (NP (NG (NN (N fading)))
                (PP (P of)
                    (NP (NG (NN (N satellite)
                                (NN (N transmissions)))))))
        (CONJ and)
        (NP (NG (ADJG (ADJ ionospheric))
                (NN (N irregularities))))))
```

In example (9), the chart parser generates a preliminary syntactic structure which disagrees with the semantic plausibilities of the coordination relationships. Since semantic constraints overrule syntactic constraints, the preliminary syntactic structure is modified so that in the final syntactic structure "fading and irregularities" are coordinated instead of "satellite transmissions and irregularities".

# 5 Results and Conclusion

We tested our hybrid architecture on 158 noun phrases which were taken from the NPL corpus and which contained the conjunction "and". The chart parser generated a preliminary parsing structure for these noun phrases based on the context-free rules and based on the lexicon which currently contains about 900 words with their syntactic categories. Within the preliminary syntactic structure several forms of coordination could be detected based on syntactic constraints alone. In 89 of the 158 noun phrases there were no coordination ambiguities because the coordination was at the beginning (e.g., "Space probes and satellites"). In 14 noun phrases the coordination was between adjectives (e.g., "Observation of single and double inflexions") and in 2 noun phrases the coordination was between explicitly repeated prepositions in prepositional phrases (e.g., "Electron collision frequencies in nitrogen and in the lower ionosphere"). The remaining 53 of the 158 noun phrases were more complex and needed semantic plausibility networks as well. Using the plausibility networks as a means to correct a preliminary syntactic structure all 40 noun phrases with coordination relationships from the training corpus and 11 of 13 noun phrases with coordination relationships from the test corpus were assigned the right structural interpretation with respect to coordination.

Our approach uses symbolic syntactic rules to generate a preliminary structure of a noun phrase and connectionist semantic constraints to modify the representation if necessary. This approach is different from other approaches since our system learns part of its semantic constraints and since the system can generalize the learned knowledge. This hybrid approach can be adopted not only for coordination problems but for other problems as well (e.g. prepositional phrase attachment, relative clause attachment, participle constructions). In all these cases, learned semantic constraints can be used to support the disambiguation of structural representations. The hybrid model relies on symbolic rules and on bottom-up knowledge learned in connectionist networks. This allows to combine *predefined* syntactic knowledge with *learned and generalized* semantic control knowledge for structural disambiguation.

# References


Dyer M.G. 1988. *Symbolic NeuroEngineering for Natural Language Processing: A Multilevel Resarch Approach.* Technical Report UCLA-AI-88-14, University of California, Los Angeles.

Frazier L., Fodor J.D. 1978. The sausage machine: A new two-stage parsing model. *Cognition 6.*

Gazdar G., Mellish C. 1989. *Natural Language Processing in LISP.* Addison Wesley, New York.

Hendler J. 1989. Marker-passing over Microfeatures: Towards a Hybrid Symbolic/Connectionist Model. *Cognitive Science 13.*

Lehnert W.G. 1988. Symbolic/Subsymbolic Sentence Analysis: Exploiting the Best of Two Worlds. COINS Technical Report 88-99, University of Massachusetts, Amherst, MA.

NASA 1985. *NASA Thesaurus.* National Aeronautics and Space Administration.

Rumelhart D.E., Hinton G.E., Williams R.J. 1986. Learning Internal Representations by Error Propagation. In: Rumelhart D.E., McClelland J.L. (Ed.) *Parallel distributed Processing Vol 1.* MIT Press, Cambridge, MA.

Sparck-Jones K., VanRijsbergen C.J. 1976. Information Retrieval Test Collections. *Journal of Documentation 32 (1).*

Winograd T. 1983. *Language as a cognitive process.* Addison Wesley, Reading, MA.

Wilks Y. 1975. An Intelligent Analyzer and Understander of English. *Communications of the ACM 18 (5).*

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
6750 Kaiserslautern
FRG

## DFKI Publikationen

Die folgenden DFKI Veröffentlichungen oder die aktuelle Liste von erhältlichen Publikationen können bezogen werden von der oben angegebenen Adresse.

## DFKI Publications

The following DFKI publications or the list of currently available publications can be ordered from the above address.

---

**DFKI Research Reports**

**RR-90-01**
*Franz Baader*: Terminological Cycles in KL-ONE-based Knowledge Representation Languages
33 pages

**RR-90-02**
*Hans-Jürgen Bürckert*: A Resolution Principle for Clauses with Constraints
25 pages

**RR-90-03**
*Andreas Dengel, Nelson M. Mattos:* Integration of Document Representation, Processing and Management
18 pages

**RR-90-04**
*Bernhard Hollunder, Werner Nutt:* Subsumption Algorithms for Concept Languages
34 pages

**RR-90-05**
*Franz Baader:* A Formal Definition for the Expressive Power of Knowledge Representation Languages
22 pages

**RR-90-06**
*Bernhard Hollunder:* Hybrid Inferences in KL-ONE-based Knowledge Representation Systems
21 pages

**RR-90-07**
*Elisabeth André, Thomas Rist:* Wissensbasierte Informationspräsentation:
Zwei Beiträge zum Fachgespräch Graphik und KI:

1. Ein planbasierter Ansatz zur Synthese illustrierter Dokumente
2. Wissensbasierte Perspektivenwahl für die automatische Erzeugung von 3D-Objektdarstellungen

24 pages

**RR-90-08**
*Andreas Dengel:* A Step Towards Understanding Paper Documents
25 pages

**RR-90-09**
*Susanne Biundo:* Plan Generation Using a Method of Deductive Program Synthesis
17 pages

**RR-90-10**
*Franz Baader, Hans-Jürgen Bürckert, Bernhard Hollunder, Werner Nutt, Jörg H. Siekmann:* Concept Logics
26 pages

**RR-90-11**
*Elisabeth André, Thomas Rist:* Towards a Plan-Based Synthesis of Illustrated Documents
14 pages

**RR-90-12**
*Harold Boley:* Declarative Operations on Nets
43 pages

**RR-90-13**
*Franz Baader:* Augmenting Concept Languages by Transitive Closure of Roles: An Alternative to Terminological Cycles
40 pages

**RR-90-14**
*Franz Schmalhofer, Otto Kühn, Gabriele Schmidt:* Integrated Knowledge Acquisition from Text, Previously Solved Cases, and Expert Memories
20 pages

**RR-90-15**
*Harald Trost:* The Application of Two-level Morphology to Non-concatenative German Morphology
13 pages

**RR-90-16**
*Franz Baader, Werner Nutt:* Adding Homomorphisms to Commutative/Monoidal Theories, or: How Algebra Can Help in Equational Unification
25 pages

**RR-90-17**
*Stephan Busemann*
Generalisierte Phasenstrukturgrammatiken und ihre Verwendung zur maschinellen Sprachverarbeitung
114 Seiten

**RR-91-01**
*Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, and Gert Smolka :*
On the Expressivity of Feature Logics with Negation, Functional Uncertainty, and Sort Equations
20 pages

**RR-91-02**
*Francesco Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, Werner Nutt:*
The Complexity of Existential Quantification in Concept Languages
22 pages

**RR-91-03**
*B.Hollunder, Franz Baader:* Qualifying Number Restrictions in Concept Languages
34 pages

**RR-91-04**
*Harald Trost*
X2MORF: A Morphological Component Based on Augmented Two-Level Morphology
19 pages

**RR-91-05**
*Wolfgang Wahlster, Elisabeth André, Winfried Graf, Thomas Rist:* Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation.
17 pages

**RR-91-06**
*Elisabeth André, Thomas Rist:* Synthesizing Illustrated Documents
A Plan-Based Approach
11 pages

**RR-91-07**
Günter Neumann, Wolfgang Finkler: A Head-Driven Approach to Incremental and Parallel Generation of Syntactic Structures
13 pages

**RR-91-08**
*Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist*
WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation
23 pages

**RR-91-09**
*Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta*
RATMAN and its Relation to Other Multi-Agent Testbeds
31 pages

**RR-91-10**
*Franz Baader, Philipp Hanschke*
A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

**RR-91-11**
*Bernhard Nebel*
Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

**RR-91-13**
*Gert Smolka*
Residuation and Guarded Rules for Constraint Logic Programming
17 pages

**RR-91-15**
*Bernhard Nebel, Gert Smolka*
Attributive Description Formalisms ... and the Rest of the World
20 pages

**RR-91-16**
*Stephan Busemann*
Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations
18 pages

---

**DFKI Technical Memos**

**TM-89-01**
*Susan Holbach-Weber:* Connectionist Models and Figurative Speech
27 pages

**TM-90-01**
*Som Bandyopadhyay:* Towards an Understanding of Coherence in Multimodal Discourse
18 pages

**TM-90-02**
*Jay C. Weber:* The Myth of Domain-Independent Persistence
18 pages

**TM-90-03**
*Franz Baader, Bernhard Hollunder:* KRIS: Knowledge Representation and Inference System -System Description-
15 pages

**TM-90-04**
*Franz Baader, Hans-Jürgen Bürckert, Jochen Heinsohn, Bernhard Hollunder, Jürgen Müller, Bernhard Nebel, Werner Nutt, Hans-Jürgen Profitlich:* Terminological Knowledge Representation: A Proposal for a Terminological Logic
7 pages

**TM-91-01**
*Jana Köhler*
Approaches to the Reuse of Plan Schemata in Planning Formalisms
52 pages

**TM-91-02**
*Knut Hinkelmann*
Bidirectional Reasoning of Horn Clause Programs: Transformation and Compilation
20 pages

**TM-91-03**
*Otto Kühn, Marc Linster, Gabriele Schmidt*
Clamping, COKAM, KADS, and OMOS: The Construction and Operationalization of a KADS Conceptual Model
20 pages

**TM-91-04**
*Harold Boley*
A sampler of Relational/Functional Definitions
12 pages

**TM-91-05**
*Jay C. Weber, Andreas Dengel and Rainer Bleisinger*
Theoretical Consideration of Goal Recognition Aspects for Understanding Information in Business Letters
10 pages

---

**DFKI Documents**

**D-89-01**
*Michael H. Malburg, Rainer Bleisinger:*
HYPERBIS: ein betriebliches Hypermedia-Informationssystem
43 Seiten

**D-90-01**
DFKI Wissenschaftlich-Technischer Jahresbericht 1989
45 pages

**D-90-02**
*Georg Seul:* Logisches Programmieren mit Feature -Typen
107 Seiten

**D-90-03**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner:* Abschlußbericht des Arbeitspaketes PROD
36 Seiten

**D-90-04**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner:* STEP: Überblick über eine zukünftige Schnittstelle zum Produktdatenaustausch
69 Seiten

**D-90-05**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner:* Formalismus zur Repräsentation von Geo-metrie- und Technologieinformationen als Teil eines Wissensbasierten Produktmodells
66 Seiten

**D-90-06**
*Andreas Becker:* The Window Tool Kit
66 Seiten

**D-91-01**
*Werner Stein , Michael Sintek*
Relfun/X - An Experimental Prolog Implementation of Relfun
48 pages

**D-91-03**
*Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause*
RFM Manual: Compiling RELFUN into the Relational/Functional Machine
43 pages

**D-91-04**
DFKI Wissenschaftlich-Technischer Jahresbericht 1990
93 Seiten

**D-91-07**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner*
TEC-REP: Repräsentation von Geometrie- und Technologieinformationen
70 Seiten

**D-91-08**
*Thomas Krause*
Globale Datenflußanalyse und horizontale Compilation der relational-funktionalen Sprache RELFUN
137 pages

**D-91-09**
*David Powers and Lary Reeker (Eds)*
Proceedings MLNLO^91 - Machine Learning of Natural Language and Ontology
211 pages

**MLNLO'91 – Machine Learning of Natural Language and Ontology**

**David Powers & Larry Reeker (Eds)**

**D-91-09**
Document