Towards Adaptive Argumentation Learning Systems

Theoretical and Practical Considerations in the
Design of Argumentation Learning Systems

Dissertation

zur Erlangung des akademischen Grades eines
Doktors der Philosophie
der Philosophischen Fakultät III
der Universität des Saarlandes

vorgelegt von

Oliver Scheuer

aus Sankt Wendel

Saarbrücken, 2015

Dekan

Prof. Dr. Roland Brünken
Universität des Saarlandes, Fachrichtung Bildungswissenschaften

Berichterstatter

Prof. Dr. Armin Weinberger
Universität des Saarlandes, Fachrichtung Bildungswissenschaften

Prof. Dr. Bruce M. McLaren
Carnegie Mellon University, Human-Computer Interaction Institute

Tag der Disputation

17. September 2015

Acknowledgements

There are certainly two sides to the creation of any intellectual piece of work such as a doctoral dissertation. On the one hand, most obviously, the creator makes a contribution to a community. In the case of a doctoral thesis, a doctoral candidate extends the body of knowledge of the scientific community he or she engages in. On the other hand, the process of creating also alters the creator him or herself in significant ways—particularly in such a prolonged and intense project like a doctorate, which gives rise to new insights, new perspectives, and new ways of thinking. A doctoral dissertation is never a purely solitary endeavor. Rather, it is to a large extent the result of the interaction of a doctoral candidate with his or her personal and academic surrounding, that is, people who provide support and guidance at all different kinds of levels, be it intellectual, motivational, technical, or administrative. I am grateful to a great number of people who did not only help making this dissertation thesis possible (and hopefully, to some degree, interesting and insightful) but also helped me grow intellectually.

First of all, I am indebted to Prof. Dr. Bruce McLaren, who offered me the opportunity to make a Ph.D., helped defining research questions that are both, interesting and important, and gave invaluable guidance and support throughout the whole endeavor. I learned and improved many essential skills through the mentoring of Bruce such as applying scientific methods in a proper way, doing research with scientific integrity and rigor, presenting ideas in a clear and compelling manner, and, last but not least, employing sound argumentation in thinking and talking. Working in his research team was always motivating and inspiring. The interaction with Bruce and others in his team was in the truest sense of the word *collaborative*, with a working atmosphere being pleasant, supportive, and productive at the same time.

Through a fortunate coincidence, one of the leading researchers on argumentation learning and scripted collaboration, Prof. Dr. Armin Weinberger, assumed a professorship of Educational Technologies at Saarland University in 2010. As my doctorate supervisor, Armin's advice was extremely helpful in defining and shaping the parts of my research that are oriented towards educational psychology. His feedback and suggestions helped me significantly improve the written thesis. It is rather unusual that a computer science graduate aspires to a Ph.D. in a social science discipline. Armin's support and engagement was instrumental in sorting out the red tape of university regulations to make this possible.

I want to thank our collaborators within the LASAD project, which the biggest part of this dissertation originates from. The accomplishments achieved within LASAD would not have been possible without such great collaborators as Prof. Dr. Niels Pinkwart and his team. Dr. Frank Loll did a tremendous job in developing the LASAD system, which was the basis of many aspects of my work. When facing technical problems, I could always rely on Frank's prompt response and action. In the last phase of the LASAD project, Sabine Niebuhr joined the Clausthal team. Sabine generously agreed to help coding chat protocols for qualitative aspects of argumentation, a laborious and time-consuming task. I also want to thank the members of the Argunaut project, and, in particular, the teams at the Hebrew University of Jerusalem and at the University of Exeter, who coded the data that I used in the machine learning experiments presented in this dissertation.

A central question in educational research is always how to access and recruit students to participate in empirical studies—a problem that sometimes turns out to be more difficult than expected. Prof. Dr. Maralee Harrell gave me the opportunity to carry out an empirical study in one of her Philosophy classes at the Carnegie Mellon University. A second empirical study—this time at Saarland University—only became possible through the generous support of Prof. Dr. Christoph Fehige. Christoph actively supported me in recruiting students over several channels such as announcements on the Philosophy mailing list and in several Philosophy lectures. Moreover, through his initiative and help, it was possible to invite Prof. Dr. Rudolf Schüßler, a renowned expert in climate ethics, to give a guest lecture to introduce the study topic to our subjects.

I want to express my gratitude to the people who supported the preparation, execution, and analysis of empirical studies. I want to thank Dr. Dimitra Tsovaltzi and Dr. Toby Dragon, with each of whom I shared an office for a considerable time and could discuss my thoughts and ideas. Dimitra and Toby also reviewed and commented on the materials of the Saarbrücken study and provided assistance during its execution. Florian Zickwolf proofread the texts used in the study. Vera Gehlen-Baum provided feedback on the instructional materials and the coding approach used to analyze the recorded discussion protocols.

Prof. Dr. Carolyn Penstein Rosé did not only make her machine learning toolkit TagHelper available to me and provided guidance in using it. She also gave advice on the empirical evaluation of machine learning approaches, which was extremely helpful in context of the Argunaut machine learning experiments. During a mentoring session at the International Conference on Computers in Education in 2011, Prof. Dr. Ulrich Hoppe provided me excellent feedback on a manuscript describing parts of my research.

A number of student workers assisted in the technical implementations presented in this dissertation. I want to highlight the contributions of Almer Bolatov, Ugur Kira, and Anahuac Valero Haro, who were always highly committed to their work and easy to work with.

Life at a university or research institution does not exclusively consist of hard work and long hours in the office. I want to thank my former colleagues of the ActiveMath group at the German Research Center for Artificial Intelligence and Saarland University. In particular, I want to thank Eric Andres, Michael Dietrich, Dr. George Goguadze, Thorsten Hey, and Dr. Sergey Sosnovsky for the many enjoyable Friday afternoons we spent playing table soccer.

In the last phase of my dissertation, it was not always easy to combine a full-time job in industry and the writing of a doctoral thesis. I want to thank the ORBIS AG, and in particular, my boss Martin Simmet, who was flexible in arranging my working times in a way that I could still fulfill my obligations at the university.

Last but not least, I want to thank my family and friends for their continuing support and encouragement, without which this work would not have happened.

Parts of this work are sponsored by the German Research Foundation (DFG) under the grant "LASAD—Learning to Argue: Generalized Support Across Domains," and by the 6th Framework Program of the European Community under the grant “Argunaut—An Intelligent Guide to Support Productive Online Dialogue,” Proposal/Contract No. 027728.

Abstract

Skills of argument play a pivotal role in many aspects of our lives. To persuade others from our opinions we require the ability to construct convincing arguments. But argumentation is not only a means of effective communication. It is also central to our thinking processes to make sound judgments and decisions. In particular, when situations are complex, when information is unclear, incomplete, and contradictory, or when conflicting values, interests, and objectives must be reconciled, it is essential to carefully weigh the arguments for and against available alternatives. We are continually facing situations of this type, be it in our jobs, in our private lives, or as citizens who form an opinion on political issues and critically and actively participate in public discussions.

Despite the fact that argumentation is one of the most fundamental skills, research consistently shows that people struggle with sound argumentative reasoning in everyday situations, and likewise, exhibit remarkable deficiencies in professional and academic argumentation. It has been criticized that our schools and universities are responsible for a large share in this disappointing result. It is said that education largely neglects this important set of skills, thus, fails not only in imparting argumentation skills as such but also in fostering a positive attitude and appreciation towards argumentation as a social practice and way of knowing.

Given this need to place more emphasis on argumentation, it is not surprising that approaches to fostering argumentation learning have been a considerable focus of educational research. In particular, state of the art computer technologies may offer new opportunities to significantly improve current practices in teaching and learning of argumentation skills. Adaptation technologies hold promise for a new generation of systems that adjust to individual differences of learners and flexibly respond to situational demands, thereby equally improving user experience and learning effectiveness. While research has achieved impressive advances over the last three decades, many crucial aspects in designing and developing effective argumentation learning technologies are still not sufficiently understood, which is a key reason preventing an uptake of such technologies on a broader scale.

This dissertation thesis seeks to make a contribution to this important research area at the intersection of computer science and the learning sciences. The thesis is the result of an interdisciplinary research program that addresses four issues of pivotal importance in realizing the promises of adaptive argumentation learning systems:

1. *User interface:* How can argumentation user interfaces be designed to effectively structure and support problem solving, peer interaction, and learning?
2. *Software architecture:* How can software architectures of adaptive argumentation learning systems be designed to be employable across different argumentation domains and application scenarios in a flexible and cost-effective manner?
3. *Diagnostics:* How can user behavior be analyzed, automatically and accurately, to drive automated adaptations and help generation?
4. *Adaptation:* How can strategies for automated adaptation and support be designed to promote problem solving, peer interaction, and learning in an optimal fashion?

Regarding issue (1), this dissertation investigates argument diagrams and structured discussion interfaces, two areas of focal interest in argumentation learning research during the past decades. The foundation for such structuring approaches is given by theories of learning and teaching with knowledge representations (*theory of representational guidance*) and collaboration scripts (*script theory of guidance in computer-supported collaborative learning*). This dissertation brings these two strands of research together and presents a computer-based learning environment that combines both approaches to support students in conducting high-quality discussions of controversial texts. The diagrams help learners prepare for the discussion by providing a tool to graphically organize, and consequently better understand, the relationships between claims and arguments of given texts. Moreover, during the discussions, the diagrams are available as graphical agendas and material collections, which stimulate new discussion contributions and to which references may be made. The structuring of the discussions through a collaboration script helps students engage in critical-constructive forms of dialog. An empirical study confirms that this combined approach has positive impact on the quality of discussions, thus, underpins the theoretical basis of the approach. Beyond that, the present approach lays the foundation for subsequent research. In particular, anecdotal evidence suggests that combining the two structuring elements may lead to positive synergistic effects, an observation that calls for future research on combining different structuring approaches more generally.

Regarding issue (2), this dissertation presents a software framework for enhancing argumentation systems with adaptive support mechanisms. Adaptive support functionality of past argumentation systems has been tailored to particular domains and application scenarios. A novel software framework is presented that abstracts from the specific demands of different domains and application scenarios to provide a more general approach. The approach comprises an extensive configuration subsystem that allows the flexible definition of intelligent software agents, that is, software components able to reason and act autonomously to help students engage in fruitful learning activities. The behavior of these agents can be configured by defining pedagogically relevant patterns, feedback messages, and feedback strategies. Four showcase applications highlight specific capabilities and design options offered by the software framework; together, they demonstrate the generality and breadth of the approach. Thus, the present approach makes an important contribution to the software design of adaptive argumentation systems. The development of adaptive learning technologies is generally complex and time-consuming and requires technical, domain-specific, and pedagogical expert knowledge. Therefore, a graphical authoring tool has been conceptualized and implemented to simplify the process of defining and administering software agents beyond what has been achieved with the provided framework system. Among other things, the authoring tool allows, for the first time, specifying relevant patterns in argument diagrams using a graphical language. Empirical results indicate the high potential of the authoring approach but also challenges for future research.

Regarding issue (3), the dissertation investigates two alternative approaches to automatically analyzing argumentation learning activities: the knowledge-driven and the data-driven analysis method. The knowledge-driven approach utilizes a pattern search component to identify relevant structures in argument diagrams based on declarative pattern specifications. The capabilities and appropriateness of this approach are demonstrated through three exemplary applications, for which pedagogically relevant patterns have been defined and implemented within the component. The approach proves particularly useful for patterns of limited complexity in scenarios with sufficient expert knowledge available. The data-driven approach is based on machine learning techniques, which have been employed to induce computational classifiers for important aspects of graphical online discussions, such as off-topic contributions, reasoned claims, and question-answer interactions. Validation results indicate that this approach can be realistically used even for complex classification tasks involving natural language. This research constitutes the first investigation on the use of machine learning techniques to analyze diagram-based educational discussions. It thus provides a solid foundation for future research and practical use of adaptation technologies in the field of computer-supported collaborative learning. All in all, the knowledge-driven and data-driven approaches to analyze argumentation learning activities can be considered complementary. The dissertation brings up and discusses the still largely unexplored topic of capitalizing on a combination of both approaches. The developed framework system offers the required technical infrastructure to explore this question in greater detail in future research and to implement corresponding practical solutions.

The dissertation concludes with discussing the four addressed research challenges in the broader context of existing theories and empirical results. The pros and cons of different options in the design of argumentation learning systems are juxtaposed; areas for future research are identified. This final part of the dissertation gives researchers and practitioners a synopsis of the current state of the art in the design of argumentation learning systems and its theoretical and empirical underpinning. Special attention is paid to issue (4), with an in-depth discussion of existing adaptation approaches and corresponding empirical results. In closing, the non-adaptive argumentation environment, which was designed and researched in context of issue (1), is revisited. Prospects for enhancing this environment with automated feedback are elaborated. In particular, ways to enable the automated analysis and support of online discussions both on the content level and the social level are discussed. A key to success may be the exploitation of machine interpretable user inputs with explicit semantics, such as typed diagram elements, pre-structured discussion contributions, and explicit references between diagram elements and discussion contributions. The dissertation presents initial insights on the feasibility of such an approach, gained from an exploratory data analysis.

Zusammenfassung

Argumentationsfähigkeiten nehmen eine herausragende Stellung in vielen Bereichen des Lebens ein. Um andere von unseren Ansichten zu überzeugen, brauchen wir die Fähigkeit, überzeugend zu argumentieren. Argumentation ist jedoch nicht nur ein Mittel zur effektiven Kommunikation, sondern spielt auch eine wesentliche Rolle innerhalb unserer Denkprozesse, um Urteile und Entscheidungen in rationaler Weise treffen zu können. Insbesondere wenn Situationen sich komplex gestalten, wenn Informationen unklar, unvollständig und widersprüchlich sind, oder wenn gegensätzliche Werte, Interessen und Ziele in Einklang zu bringen sind, müssen Argumente und Gegenargumente sorgfältig abgewogen werden. Wir stehen solchen Situationen ständig gegenüber, sei es in unserem Beruf, in unserem Privatleben oder auch als Bürger, die sich zu politische Fragen eine Meinung bilden und sich kritisch und aktiv an öffentlichen Diskussionen beteiligen.

Nichtsdestotrotz zeigen Forschungsergebnisse übereinstimmend, dass bemerkenswerte Schwächen sowohl bei informellen, beruflichen als auch akademischen Formen der Argumentation weit verbreitet sind. Einen großen Anteil an diesem Missstand wird unseren Schulen und Universitäten zugeschrieben, welche die Ausbildung von Argumentationskompetenzen sträflich vernachlässigten. Damit werde nicht nur die Vermittlung von Argumentationsfähigkeiten an sich versäumt, sondern ebenso die Förderung einer positiven Einstellung und Wertschätzung gegenüber Argumentation als soziale Praxis und Erkenntnismethode.

Es ist daher nicht weiter überraschend, dass die Bildungsforschung Ansätze zum Erlernen von Argumentationsfähigkeiten verstärkt in den Blick genommen hat. Insbesondere moderne Computertechnologien bieten neue Möglichkeiten, die aktuelle Praxis des Lehrens und Lernens von Argumentationsfähigkeiten in entscheidender Weise zu verbessern. Adaptionstechnologien versprechen eine neue Generation von Systemen, welche sich an individuelle Unterschiede von Lernenden anpassen und flexibel auf situative Anforderungen reagieren, um gleichermaßen Nutzererlebnis und Lernerfolg zu verbessern. Obwohl die Forschung in den letzten drei Jahrzehnten beeindruckende Fortschritte erreicht hat, sind wesentliche Aspekte hinsichtlich des Designs effektiver Lerntechnologien für Argumentation immer noch nicht hinreichend verstanden, was einem Einsatz solcher Systeme auf breiterer Basis entgegensteht.

Diese Dissertationsschrift widmet sich diesem wichtigen Bereich in der Schnittmenge zwischen informatischer und bildungswissenschaftlicher Forschung. Sie behandelt im Rahmen eines interdisziplinären Forschungsprogramms die folgenden vier Fragestellungen, welche bei der Realisierung adaptiver Argumentationssysteme von zentraler Bedeutung sind:

1. *Benutzerschnittstelle:* Wie müssen Benutzerschnittstellen beschaffen sein, um Problemlöse-, Kooperations- und Lernprozesse effektiv zu strukturieren und zu unterstützen?
2. *Softwarearchitektur:* Wie können die Funktionalitäten eines adaptiven Argumentationslernsystems in eine Softwarearchitektur abgebildet werden, welche flexibel und mit angemessenem Aufwand in verschiedenen Bereichen und Szenarien einsetzbar ist?
3. *Diagnostik:* Wie kann Benutzerverhalten automatisch und mit hoher Genauigkeit analysiert werden, um automatisierte Anpassungen und Hilfestellungen effektiv zu steuern?
4. *Adaption:* Wie sollten automatisierte Anpassungen und Hilfestellungen ausgestaltet werden, um Problemlöse-, Kooperations- und Lernprozesse optimal zu unterstützen?

Hinsichtlich Fragestellung (1) untersucht diese Arbeit Argumentationsdiagramme und strukturierte Onlinediskussionen, zwei Schwerpunkte der Forschung zu Lernsystemen für Argumentation der vergangenen Jahre. Die Grundlage solcher Strukturierungsansätze bilden Theorien zum Lehren und Lernen mit Wissensrepräsentationen (*theory of representational guidance*) und Kooperationsskripten (*script theory of guidance in computer-supported collaborative learning*). Diese Arbeit führt beide Forschungsstränge in einer neuartigen Lernumgebung zusammen, die beide Ansätze vereint, um Lernende beim Diskutieren kontroverser Texte zu unterstützen. Die Diagramme dienen, zum einen, zur Diskussionsvorbereitung, um die Beziehungen zwischen den Aussagen und Argumenten der Texte zu analysieren und dadurch besser zu verstehen. Zum anderen stehen die Diagramme während der Diskussion als grafische Agenden und Stoffsammlungen zur Verfügung, welche Diskussionsbeiträge anregen und auf die Bezug genommen werden kann. Die Strukturierung der Diskussionen mittels eines Kooperationsskriptes unterstützt Lernende in kritisch-konstruktiven Formen des Dialogs. Eine empirische Untersuchung zeigt, dass sich dieser kombinierte Ansatz positiv auf die Diskussionsqualität auswirkt und bekräftigt damit die zu Grunde liegenden theoretischen Annahmen. Der vorgestellte Ansatz legt darüber hinaus den

Grundstein für weiterführende Forschung. Es ergaben sich insbesondere vielversprechende Anhaltspunkte für positive Effekte der Verzahnung beider Strukturierungselemente, was zur Frage der Kombination verschiedener Strukturierungsansätze im Allgemeinen führt.

Hinsichtlich Fragestellung (2) stellt diese Arbeit ein Software-Rahmensystem zur Bereitstellung adaptiver Unterstützungsmechanismen in Argumentationssystemen vor. Während die Unterstützungsmechanismen vorheriger Argumentationssysteme auf bestimmte Domänen und Anwendungsszenarien zugeschnitten waren, abstrahiert das hier vorgestellte Rahmensystem von domänen- und anwendungsspezifischen Besonderheiten und stellt damit einen generelleren Ansatz dar. Der Ansatz umfasst ein umfangreiches Konfigurationssystem zur Definition intelligenter Softwareagenten, d. h. Softwarekomponenten, die eigeständig schlussfolgern und handeln, um Lernprozesse zu unterstützen. Die Konfiguration des Verhaltens der Agenten erfolgt durch die Festlegung pädagogisch relevanter Muster, Rückmeldungsnachrichten und Rückmeldungsstrategien. Vier beispielhafte Anwendungen heben bestimmte Fähigkeiten und Möglichkeiten des Rahmensystems hervor; in ihrer Gesamtheit demonstrieren sie die Allgemeinheit und Breite des Ansatzes. Damit liefert diese Arbeit einen wichtigen Beitrag zur Frage der softwaretechnischen Gestaltung adaptiver Argumentationssysteme. Das Erstellen adaptiver Lernsysteme ist im Allgemeinen komplex und zeitaufwändig und erfordert sowohl technisches, inhaltliches als auch pädagogisches Fachwissen. Um das Definieren und Administrieren von Softwareagenten über das bereitgestellte Rahmensystem hinaus zu vereinfachen, wurde ein grafisches Autorenwerkzeug konzipiert und entwickelt. Unter anderem erlaubt dieses erstmals, relevante Muster in Argumentationsdiagrammen ohne Programmierung mittels einer grafischen Sprache zu spezifizieren. Empirische Befunde zeigen neben dem hohen Potential des Ansatzes auch die Notwendigkeit weiterführender Forschung.

Hinsichtlich Fragestellung (3) untersucht diese Arbeit zwei alternative Ansätze zur automatisierten Analyse von Lernaktivitäten im Bereich Argumentation: die wissensbasierte und die datenbasierte Analysemethodik. Der wissensbasierte Ansatz wurde mittels einer Softwarekomponente zur Mustersuche in Argumentationsdiagrammen umgesetzt, welche auf Grundlage deklarativer Musterbeschreibungen arbeitet. Die Möglichkeiten und Eignung des Ansatzes werden anhand von drei Beispielszenarien demonstriert, für die verschiedenartige, pädagogisch relevante Muster innerhalb der entwickelten Softwarekomponente definiert wurden. Der Ansatz erweist sich insbesondere als nützlich für Muster eingeschränkter Komplexität in Szenarien, für die Expertenwissen in ausreichendem

Umfang verfügbar ist. Der datenbasierte Ansatz wurde mittels maschineller Lernverfahren umgesetzt. Mit deren Hilfe wurden Klassifikationsroutinen zur Analyse zentraler Aspekte von Onlinediskussionen, wie beispielsweise themenfremde Beiträge, begründete Aussagen und Frage-Antwort-Interaktionen, algorithmisch hergeleitet. Validierungsergebnisse zeigen, dass sich dieser Ansatz selbst für komplexe Klassifikationsprobleme eignet, welche die Berücksichtigung natürlicher Sprache erfordern. Dies ist die erste Arbeit zum Einsatz maschineller Lernverfahren zur Analyse von diagrammbasierten Lerndiskussionen. Der vorgestellte Ansatz liefert damit wichtige Grundlagen zur weiteren Erforschung und praktischen Umsetzung von Adaptionsmechanismen im Bereich des computerbasierten kooperativen Lernens. Insgesamt können der wissensbasierte und der datenbasierte Analyseansatz als komplementär betrachtet werden. Mit der Möglichkeit, beide Ansätze sinnvoll zu kombinieren, diskutiert diese Arbeit eine wichtige, bisher weitgehend unerforschte Fragestellung. Das entwickelte Rahmensystem bietet die notwendige technische Infrastruktur, diese Frage zukünftig tiefgehender zu untersuchen und entsprechende Lösungen praktisch umzusetzen.

Die Arbeit schließt mit einer Diskussion des aktuellen Forschungsstands hinsichtlich der vier Fragestellungen im breiteren Kontext existierender Theorien und empirischer Befunde. Die Vor- und Nachteile verschiedener Optionen für die Gestaltung von Lernsystemen für Argumentation werden gegenübergestellt und zukünftige Forschungsfelder vorgeschlagen. Dieser letzte Teil der Arbeit bietet Forschern und Anwendern einen umfassenden Überblick des aktuellen Forschungsstands bezüglich des Designs computerbasierter Argumentationslernsysteme und den zugrunde liegenden lehr- und lerntheoretischen Erkenntnissen. Insbesondere wird auf Fragestellung (4) vertiefend eingegangen und bisherige Adaptionsansätze einschließlich entsprechender empirischer Befunde erörtert. Schließlich wird näher auf Möglichkeiten eingegangen, die unter Fragestellung (1) entwickelte Argumentationsumgebung mit automatisiertem Feedback zu erweitern. Es werden mögliche Ansätze zur automatischen Analyse und Unterstützung von Onlinediskussionen sowohl auf der inhaltlichen als auch der sozialen Ebene diskutiert. Ein Schlüssel zum Erfolg könnte in der Nutzung maschinell interpretierbarer Benutzereingaben mit expliziter Semantik liegen, wie beispielsweise typisierte Diagrammelemente, vorstrukturierte Diskussionsbeiträge oder explizit vom Benutzer erstellte Bezüge zwischen Diagrammelementen und Diskussionsbeiträgen. Erste Einblicke in die Umsetzbarkeit dieser Optionen werden anhand der Ergebnisse einer Erkundungsstudie erläutert.

Published Materials

Parts of this dissertation thesis have already been published in the following research papers:

McLaren, B. M., Scheuer, O., De Laat, M., Hever, R., De Groot, R., & Rosé, C. (2007). Using machine learning techniques to analyze and support mediation of student e-discussions. In R. Luckin, K. R. Koedinger, & J. Greer (Eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (pp. 331–340), IOS Press, Amsterdam.

Scheuer, O., & McLaren, B. M. (2008). Helping teachers handle the flood of data in online student discussions. In B. P. Woolf, E. Aimeur, R. Nkambou, & S. Lajoie (Eds.), *Proceedings of the 9th International Conference in Intelligent Tutoring Systems, LNCS 5091* (pp. 323–332), Springer, Berlin.

McLaren, B. M., Scheuer, O., & Mikšátko, J. (2010). Supporting collaborative learning and e-discussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education, 20*(1), 1–46. doi:10.3233/JAI-2010-0001

Scheuer, O., Loll, F., Pinkwart, N., & McLaren, B. M. (2010). Computer-supported argumentation: A review of the state-of-the-art. *International Journal of Computer-Supported Collaborative Learning, 5*(1), 43–102. doi:10.1007/s11412-009-9080-x

Scheuer, O., McLaren, B. M., Harrell, M., & Weinberger, A. (2011). Scripting collaboration: What affect does it have on student argumentation? In T. Hirashima et al. (Eds.), *Proceedings of the 19th International Conference on Computers in Education* (pp. 181–188). Asia-Pacific Society for Computers in Education, Chiang Mai, Thailand.

Scheuer, O., McLaren, B. M., Harrell, M., & Weinberger, A. (2011). Will structuring the collaboration of students improve their argumentation? In G. Biswas, S. Bull, J. Kay, & A. Mitrovic (Eds.), *Proceedings of the 15th International Conference on Artificial Intelligence in Education, LNAI 6738* (pp. 544–546), Springer, Berlin.

Scheuer, O., McLaren, B. M., Loll, F., & Pinkwart, N. (2012). Automated analysis and feedback techniques to support and teach argumentation: A survey. In N. Pinkwart, & B. M. McLaren (Eds.), *Educational technologies for teaching argumentation skills* (pp. 71–124). Bentham Science Publishers, Sharjah, UAE.

Scheuer, O., Niebuhr, S., Dragon, T., McLaren B. M., & Pinkwart, N. (2012). Adaptive support for graphical argumentation: The LASAD approach. *IEEE Learning Technology Newsletter 14*(1), 8–11.

Scheuer, O., & McLaren, B. M. (2013). CASE: A configurable argumentation support engine. *IEEE Transactions on Learning Technologies, 6*(2), 144–157. doi:10.1109/TLT.2013.3

Scheuer, O., McLaren, B. M., & Weinberger, A. (2013). Supporting discussions through argument diagrams and collaboration scripts. Presented as part of the symposium "Computer-Supported

Collaborative Learning (CSCL): Exploring Synergetic Scaffolding and Scripting" at the *15th Biennial EARLI Conference for Research on Learning and Instruction*.

Scheuer, O., McLaren, B. M., & Weinberger, A. (2013). Automated and adaptive support for educational discussions: Results to guide in making this a reality. In N. Rummel, M. Kapur, M. Nathan, & S. Puntambekar (Eds.), *To See the World and a Grain of Sand: Learning across Levels of Space, Time, and Scale: CSCL 2013 Conference Proceedings Volume 2 – Short Papers, Panels, Posters, Demos & Community Events* (pp. 349–350). International Society of the Learning Sciences.

Scheuer, O., McLaren, B. M., Weinberger, A., & Niebuhr, S. (2014). Promoting critical, elaborative discussions through a collaboration script and argument diagrams. *Instructional Science, 42*(2), 127–157. doi:10.1007/s11251-013-9274-5

Primary Research Articles

The following research articles constitute the foundation of this dissertation thesis:

Article 1

Scheuer, O., Loll, F., Pinkwart, N., & McLaren, B. M. (2010). Computer-supported argumentation: A review of the state-of-the-art. *International Journal of Computer-Supported Collaborative Learning, 5*(1), 43–102. doi:10.1007/s11412-009-9080-x

Article 2

Scheuer, O., McLaren, B. M., Weinberger, A., & Niebuhr, S. (2014). Promoting critical, elaborative discussions through a collaboration script and argument diagrams. *Instructional Science, 42*(2), 127–157. doi:10.1007/s11251-013-9274-5

Article 3

Scheuer, O., & McLaren, B. M. (2013). CASE: A configurable argumentation support engine. *IEEE Transactions on Learning Technologies, 6*(2), 144–157. doi:10.1109/TLT.2013.3

Article 4

McLaren, B. M., Scheuer, O., & Mikšátko, J. (2010). Supporting collaborative learning and e-discussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education, 20*(1), 1–46. doi:10.3233/JAI-2010-0001

Table of Contents

List of Figures

Background (Chapter 1 – 3)

Supporting Argumentation Learning through Multilevel Scaffolding (Chapter 4)

Designing a Configurable Argumentation Support Engine (Chapter 5)

Using Machine Learning Techniques to Analyze Student Discussions (Chapter 6)

List of Tables

Background (Chapter 1 – 3)

Supporting Argumentation Learning through Multilevel Scaffolding (Chapter 4)

Designing a Configurable Argumentation Support Engine (Chapter 5)

Using Machine Learning Techniques to Analyze Student Discussions (Chapter 6)

Synthesis (Chapter 7 – 10)

Overview

It is widely recognized that critical thinking skills play an important role in today's knowledge societies. Being able to understand and critically evaluate arguments of others, and, likewise, to produce arguments in a well-reasoned way, is crucial in many professions (e.g., science, the law, politics), in the private sphere, and in participating in democratic societies. Despite the ubiquity and importance of argumentation skills across many aspects of human life, the educational system largely fails to pay due respect to this important set of skills (Driver et al. 2000; Osborne 2010; Sampson and Blanchard 2012; Simon et al. 2006). As a result, research on professional and everyday reasoning consistently shows that people struggle in producing sound and valid arguments, judging the quality of arguments of others, weighing arguments to make well-reasoned decisions, and engaging in productive forms of collaborative argumentation (e.g., Kuhn 1991; Stark and Krause 2006; Weinberger and Fischer 2006).

Computer-based tools specifically designed to leverage the quality of argumentation have the potential to make an important contribution to improve this situation. During the past three decades, many such tools have been developed to support argumentation, the acquisition of argumentation skills, and learning through argumentation (overviews are provided in: Andriessen et al. 2003; Kirschner et al. 2003; Scheuer et al. 2010; Schneider et al. 2013). Although much progress has been made in recent years, there are still many open research questions regarding the design of such tools. The building of educational argumentation systems is generally not a trivial matter, since technological, pedagogical, and human-computer interaction aspects must be taken into account in creating tools that effectively support argumentation learning. In particular, the question of how systems can guide and support the learning process of groups of learners has become an important research focus. Based on the body of empirical results available so far, *theories of guidance in computer-supported collaborative learning (CSCL)* have been proposed (Fischer et al. 2013; Suthers 2003). Although these theories constitute important theoretical groundwork, new questions are now emerging, calling for further empirical investigation and theory refinement. One logical next step is to further our knowledge of how different guidance approaches can be fruitfully combined—whether the different approaches complement, reinforce, or inhibit one another (Tabak 2004). Perhaps the most visionary research challenge is whether collaborative learning systems can provide guidance in an *adaptive* fashion (Fischer et al. 2013). While a relatively comprehensive body of research exists for intelligent tutoring systems (e.g., VanLehn 2006; Woolf 2008)—which traditionally focus on individual learners—corresponding research for collaborative learning systems is still sparse, let alone a systemic and comprehensive theory of adaptive guidance in CSCL.

This dissertation explores key aspects in the design of argumentation learning systems. A special focus is placed on the two issues mentioned above, combining different forms of guidance and imbuing systems with adaptivity mechanisms. The present work is based on a series of peer-reviewed journal articles that have been written as part of the dissertation project. Besides re-presenting the content of these articles, the dissertation contains a considerable number of additional illustrations, research results, and conclusions not published yet. The specific research contributions are preceded by an extensive discussion of the relevant background to clarify the intellectual roots and theoretical foundations of the present work. Finally, the individual findings and conclusions are related to one another and discussed in the broader context of existing theories and empirical results.

The work presented in this dissertation is based on an interdisciplinary research program, which equally draws from and contributes to the learning sciences (in particular, learning with knowledge representations and collaboration scripts) and computer science research (in particular, adaptive software systems, rule-based analysis, and machine learning). While the computer science part is more applied in nature, with practical contributions to our knowledge base of how to design adaptive learning technologies, the learning science part builds upon and advances current theoretical frameworks of learning, more specifically, the theory of representational guidance (Suthers 2003) and the currently emerging script theory of guidance in computer-supported collaborative learning (Fischer et al. 2013).

Figure 1 illustrates the overall structure of the dissertation thesis. The blue shaded area shows the overarching question and guiding theme—*How to design adaptive argumentation learning systems?*—together with the four central sub-aspects investigated as part of the dissertation. In particular, I have researched questions regarding argumentation user interfaces, software architectures for adaptive support, approaches to the automated analysis of argumentation, and approaches to the adaptive support of argumentation-based learning activities. The yellow shaded areas represent the three main parts of the dissertation, each consisting of several chapters. The red solid arrows indicate to which research aspect the different parts and chapters primarily contribute (contributions to other aspects may exist as well yet to a lesser extent). The chapters of Part B, *Research Components*, mainly focus on three sub-aspects: the design of argumentation learning user interfaces (Chapter 4), software architectures to adaptively support argumentation learning (Chapter 5), and approaches to automatically analyze argumentation (Chapter 5 and Chapter 6). The chapters of Part A, *Background*, and Part C, *Research Synthesis*, do essentially equally contribute to all research aspects, with the exception of Chapter 9, which is mainly concerned with ways to adaptively support argumentation learning activities. Red boxes in Figure 1 represent published essays in which parts of the research discussed have been reported.

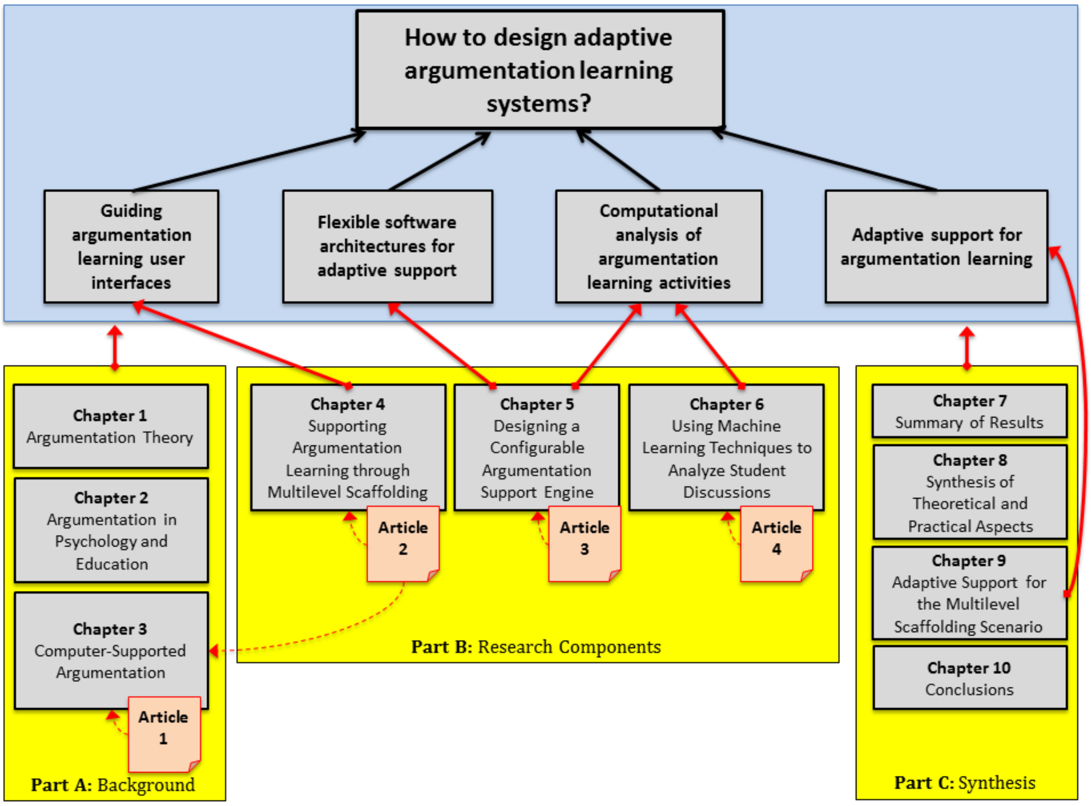

How to design adaptive argumentation learning systems?
Guiding argumentation learning user interfaces
Flexible software architectures for adaptive support
Computational analysis of argumentation learning activities
Adaptive support for argumentation learning
Chapter 1 Argumentation Theory
Chapter 2 Argumentation in Psychology and Education
Chapter 3 Computer-Supported Argumentation
Article 1
Part A: Background
Chapter 4 Supporting Argumentation Learning through Multilevel Scaffolding
Article 2
Chapter 5 Designing a Configurable Argumentation Support Engine
Article 3
Chapter 6 Using Machine Learning Techniques to Analyze Student Discussions
Article 4
Part B: Research Components
Chapter 7 Summary of Results
Chapter 8 Synthesis of Theoretical and Practical Aspects
Chapter 9 Adaptive Support for the Multilevel Scaffolding Scenario
Chapter 10 Conclusions
Part C: Synthesis

Figure 1: Structure of the dissertation thesis.

Part A discusses the research background of this dissertation by reviewing argumentation from different disciplinary angles. Chapter 1 starts with an overview of philosophical treatments of argumentation and linguistic studies of discourse. Corresponding theories provide a conceptual framework of how argumentation can be conceived of and help understand how argumentation unfolds in real-world conversations. This foundational work had, and still has, a strong influence on research on argumentation in the learning sciences. Chapter 2 reviews argumentation research in psychology and education, covering crucial aspects such as the skills of argument, cognitive and social theories of argumentation, the development of argumentation skills, and educational approaches to supporting the learning of and through argumentation. A special focus is placed on social learning theories and their relation and application to argumentation. Chapter 3 covers computer-based approaches to supporting argumentation learning. The main focus of this chapter is research undertaken in the areas of computer-supported collaborative learning and intelligent tutoring systems. The chapter draws from a review of the state of the art of computer-based argumentation systems, which was originally published in Article 1 [Scheuer, Loll, Pinkwart, and McLaren (2010)]. The results are based on an extensive review of published literature and available systems, covering more than 50 computer-based argumentation systems developed over the past 25 years. The article has been well appreciated by the research community with 198 citations by June 14, 2015, according to Google Scholar (7th most cited article of the *International Journal of Computer-Supported Collaborative Learning (IJCSCL)*, note that all articles with more citations were published earlier).[1] Additional crucial information not available in published work was elicited directly from researchers involved in the development of those systems. Other relevant input to this chapter comes from the background sections of Article 2 [Scheuer, McLaren, Weinberger, and Niebuhr (2013)], which provide an in-depth discussion of two particularly relevant methods to support argumentation learning: argument diagramming and scripted discussions. The chapter also reviews corresponding theories of guidance through representational support and collaboration scripts.

Part B presents the specific research components of this dissertation. Chapter 4 discusses aspects relevant to the design of user interfaces of argumentation learning systems. It draws from Article 2 [Scheuer, McLaren, Weinberger, and Niebuhr (2014)], which reports on an empirical study that investigated an approach to scaffold student interaction and learning through a specifically designed user interface. This user interface combines two existing and already successfully used

[1] Google Scholar web address: http://scholar.google.com; query URL for top cited publication in IJCSCL: http://scholar.google.com/scholar?as_publication=international+journal+of+computer-supported+collaborative+learning

methods—argument diagrams and discussion scripts—to optimally scaffold reasoning and discussion on both the content level (i.e., support for better understanding subject matter content) and the social level (i.e., support for productive forms of interaction among students). The user interface is designed based upon theoretical assumptions regarding the effects of graphical argument representations and collaboration scripts on the quality of problem-solving, collaboration, and learning, and can be seen as a vehicle for empirically testing elements of its underlying theories. The study compares a treatment condition, in which students used a discussion script in addition to argument diagrams, with a control condition, in which students used argument diagrams but no discussion script. The results show that the combination of both methods was effective in terms of higher-quality discussions (measured through the analysis of the discussion protocols) and improved argumentation learning (according to students' self-assessments after the intervention). On the other hand, there was no difference in terms of acquired factual knowledge about covered subject matter. Overall, the study contributes to the shaping of a new theory of scripted collaborative learning. First, the analysis confirms previous results achieved with collaboration scripts in a new instructional settings. Second, an exploration of the collected data suggests that representational and script-based scaffolds may be systematically attuned to one another to achieve synergistic effects on the quality of discussion and collaboration processes. This observation raises the more general question of the interplay of differently targeted scripting elements, a topic only marginally explored (Schellens and Fischer 2013; Tabak 2004).

Chapter 5 focuses on software architectures for the automated analysis and support of argumentation learning. The presented research mainly draws from Article 3 [Scheuer and McLaren (2013)]. It starts with a short introduction to the *LASAD* project (*Learning to argue: Generalized support across domains*),[2] which provides the context of the presented research. The LASAD project aimed at developing a generalized software framework for argumentation systems, which allows building specific argumentation learning applications for different domains and learning arrangements with relatively little effort. The LASAD system played a twofold role with respect to our research. First, the development of the LASAD system itself raised a number of interesting research questions regarding the design of computer-based learning systems. Second, by its capacity to provide guidance on

[2] LASAD was a research project funded by the German Research Foundation (DFG). Project runtime: November 1, 2008 – October 31, 2012. Principle investigators: Prof. Dr. Bruce M. McLaren (then: DFKI / Saarland University) and Prof. Dr. Niels Pinkwart (then: Clausthal University of Technology).

several levels (diagrams, sentence openers, and adaptive feedback), the LASAD system was an ideal vehicle to investigate research questions in the learning sciences. My colleague Dr. Frank Loll designed and developed the basic LASAD system, a highly configurable, web-based, multi-user argumentation learning system, which is described in his doctoral dissertation (Loll 2012). I extended and used the basic LASAD system for the empirical study mentioned above (and reported in detail in Chapter 4). Furthermore, I designed and developed a software framework for adaptive support called *CASE* (*Configurable Argumentation Support Engine*), which can connect to the basic LASAD system and deploy learning-support agents to sessions in LASAD. Analogously to the design of the basic LASAD system, one of the main concerns was to achieve a broad applicability across different argumentation domains and learning settings. This goal has been achieved through a highly configurable and extensible software architecture, which allows the definition of learning-support agents and their deployment to sessions in LASAD. The software framework entails a configurable component to automatically analyze argument diagrams. This component utilizes handcrafted analysis rules, defined by human experts, to identify pedagogically relevant patterns in argument diagrams, e.g., circular arguments. The generality and breadth of applicability of the CASE engine is demonstrated through four showcase applications, each illustrating specific capabilities of the CASE engine. This chapter finally discusses the design and evaluation of a novel graphical authoring tool to facilitate the implementation of adaptive support.

Chapter 6 also addresses the automated analysis of argumentation learning but focuses on a different analysis approach. While the approach in Chapter 5 is based on patterns defined by human experts, the approach in Chapter 6 makes use of computer algorithms to automatically extract relevant patterns from existing data. The chapter summarizes relevant parts of Article 4 [McLaren, Scheuer, and Mikšátko (2010)], which describes how artificial intelligence techniques can be used to automatically analyze qualitative aspects of student discussions (e.g., off-topic contributions, question-answer sequences). The described work was conducted in the context of the *Argunaut* project (De Groot et al. 2007).[3] The main goal in Argunaut was to develop a computer-based environment to support moderators of electronic educational discussions through awareness displays, alerts, and intervention functions. Discussions in Argunaut were based on a graphical format, similar to argument diagrams in LASAD. That is, students posted new messages by creating

[3] Argunaut was a multinational research and development project funded by the European Commission (FP6-IST). Project runtime: December 1, 2005 – August 31, 2008.

boxes in a shared workspace, and replied to contributions of others by connecting their response boxes to existing boxes through graphical links. Under the supervision of Prof. Dr. Bruce McLaren, two approaches to automatically identify salient patterns in the emerging argument graphs have been developed. While my colleague Jan Mikšátko developed a novel AI-based graph-matching algorithm, I carried out machine learning experiments to induce discussion classifiers from coded data. I achieved favorable evaluation results on six coding dimensions (e.g., topic focus, reasoned claims), and integrated these classifiers later into the LASAD system. Chapter 6 discusses the machine learning experiments in greater detail.

Finally, Part C discusses and synthesizes the findings presented in the preceding chapters. Chapter 7 summarizes the main results of the individual research components. Chapter 8 systematically analyzes different options in the design of argumentation systems with respect to the user interface, automated analysis, and adaptive support. In particular, the pros and cons of different design options are discussed, considering potential pedagogical benefits and limitations, the feasibility of implementation, development costs and risks, and scope and conditions of applicability. The overarching goal of Chapter 8 is to give a synopsis of the field with a special focus on opportunities and challenges in imbuing argumentation learning systems with adaptive support mechanisms. Notably, the different design options are not independent but partly necessitate and facilitate one another. The presented analysis identifies and discusses such dependencies as well. In particular, the question is addressed how a prestructuring of user inputs at the level of the user interface can be exploited to improve and enhance an automated analysis of student activities. Chapter 9 takes this ideas one step further. It describes a scenario that brings the different pieces of the dissertation thesis together in a concrete way. Building upon the learning environment presented in Chapter 4, which comprises argument diagraming activities and scripted discourse, Chapter 9 sketches how the so obtained structured user inputs can be used to enable or facilitate automated analyses of discussion contents and processes, and gives hints how this information can be utilized to provide adaptive support to students. Based on an exploratory data analysis, possible obstacles in implementing such an approach are identified and ways to overcome potential problems discussed. Finally, Chapter 10 presents the main conclusions that can be drawn regarding the investigated research questions and proposes how the field can be advanced more generally.

Part A

Background

Chapter 1
Argumentation Theory and the Study of Discourse

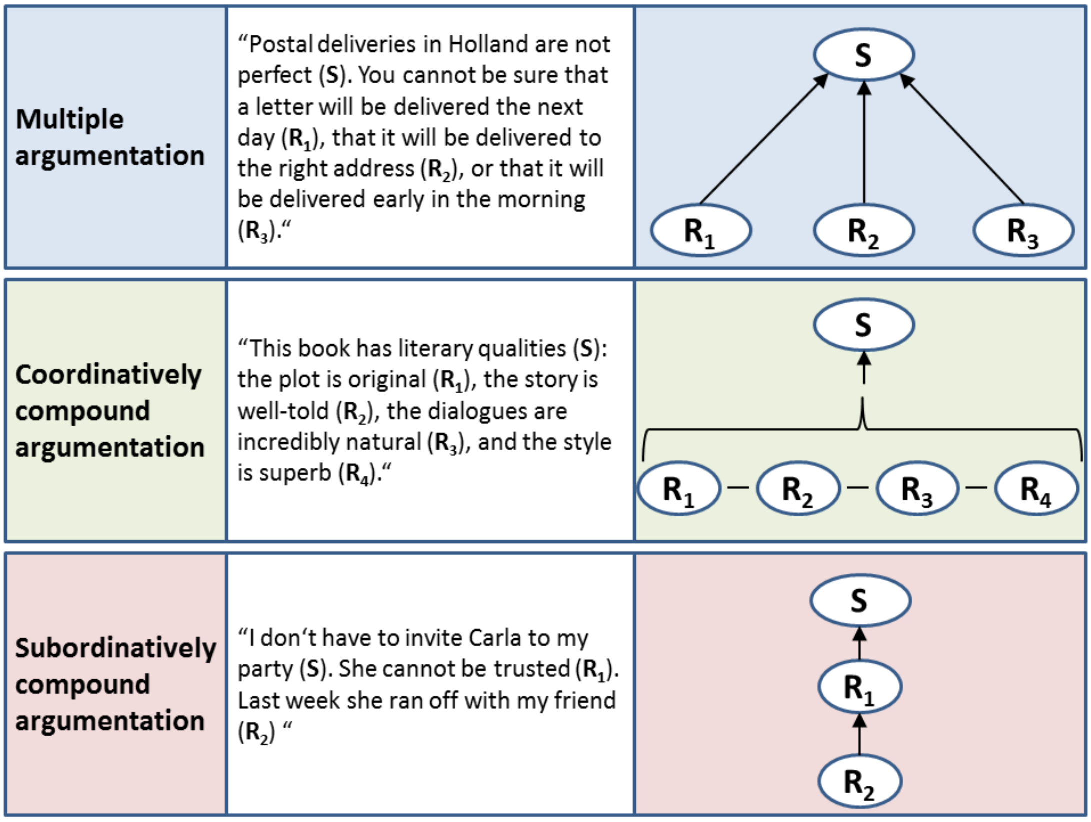
Argumentation theory is an interdisciplinary academic field concerned with the study of argumentation, with contributions of scholars from disciplines such as philosophy, linguistics, psychology, political science, communication, artificial intelligence, and the law. Argumentation theorists are interested in the production, interpretation, and evaluation of both written and oral arguments (Van Eemeren and Grootendorst 2004, p. 2). Argumentation theories lay the conceptual foundation of many approaches in educational argumentation research.

This chapter sets out with a description of the basic notions in the study of arguments (section 1.1). Then, a brief overview of relevant argumentation-theoretical groundwork is given, from Aristotle's studies of argumentation (section 1.2), which shape the understanding of argumentation to the present day, to the formal logical approach (section 1.3), which was dominant until the middle of the 20th century, to seminal work by Toulmin (1958; section 1.4), which, among other things, brought forth his still widely used model of argumentation. Section 1.5 discusses relevant contributions in philosophy and linguistics on the practical use of language in discourse, such as Grice's principles for cooperativeness in conversations, speech act theory, and the theory of grounding in communication. These lines of research have had a large impact on newer developments in argumentation theory and a direct bearing on educational research on discussion and argumentation. Section 1.6 and section 1.7 discuss influential current argumentation-theoretical work by Van Eemeren and Grootendorst (2003; section 1.6) and Walton (2008; section 1.7), which take a strong pragmatic stance in the study of argumentation. Finally, section 1.8 summarizes the main points of this chapter.

1.1 Basic Notions in the Study of Argumentation

Broadly defined, an argument can be described as any attempt to increase or decrease the acceptability of some standpoint or position. Depending on the specific theoretical stance and analytical goal, one may further restrict this definition, e.g., by requiring arguments to be rational or verbal, or emphasizing the social and/or pragmatic context of argumentation. The first part of this section focuses on the basic framework and vocabulary used by argument theorists to describe the structure and organization of arguments, drawing from an introduction given in Van Eemeren et al. (1996). The second part is concerned with categories typically employed when evaluating the quality of arguments. This part discusses the basic terminology taught in introductory philosophy courses on logic, which can be found in corresponding text books (e.g., Hurley 2008).

An *argument* comprises a *conclusion*, which is the statement one is arguing for (or against), and a set of *premises*, which are statements used to support (or oppose) the conclusion. Conclusions and premises of arguments may be explicitly stated or implicitly assumed, which requires *re-constructing* the argument in its full and explicit form when analyzing it.



Multiple argumentation

“Postal deliveries in Holland are not perfect (**S**). You cannot be sure that a letter will be delivered the next day (**R_1**), that it will be delivered to the right address (**R_2**), or that it will be delivered early in the morning (**R_3**).“

Coordinatively compound argumentation

“This book has literary qualities (**S**): the plot is original (**R_1**), the story is well-told (**R_2**), the dialogues are incredibly natural (**R_3**), and the style is superb (**R_4**).“

Subordinatively compound argumentation

“I don‘t have to invite Carla to my party (**S**). She cannot be trusted (**R_1**). Last week she ran off with my friend (**R_2**) “

Figure 2: Basic argumentation structures: Multiple (independent) argumentation (top), coordinatively compound argumentation (middle), subordinatively compound argumentation (bottom). Adapted from Van Eemeren et al. (1996, pp. 17–19).

An *argumentation* in its most basic form comprises only one pro or con argument (*single argumentation*). More complex argumentations consist of multiple pro or con arguments, which may be independent of one another (*multiple* or *convergent argumentation*, see Figure 2, top), mutually reinforce one another (*coordinatively compound* or *linked argumentation* see Figure 2, middle), or arranged in a chain of reasons (*subordinatively compound* or *serial argumentation* see Figure 2, bottom).

Individual single arguments can also be analyzed regarding the specific type of inference made. *Argumentation schemes* are general inference patterns that can be used to classify arguments in broader categories. For instance, an argument may be based on an expert opinion, empirical evidence, or an analogy. Many real-world arguments satisfy expectations regarding rationality only at their face value. If one looks deeper, one recognizes serious deficiencies that may have detrimental effects on the quality of argumentative discourse. Such deficient arguments are called *fallacies*, which constitute an important topic of research in argumentation theory. For instance, a well-known fallacy is the fallacy of *circular arguments*, also called the fallacy of *begging the question*. Argumentation schemes and fallacies are discussed in more detail below in context of Douglas Walton's work.

Besides questions of structure and organization of arguments, another important topic is the evaluation of arguments, that is, how to judge the quality of inference and the acceptability of the conclusion. Standard text books on introductory philosophy (see, for instance, Hurley 2008, one of the leading text books) typically use the following (or similar) definitions: A *deductive argument* is an argument that entails the claim that the conclusion follows necessarily from the premises. If the argument has a logical form that this claim is actually true (i.e., the conclusion does actually follow necessarily from the premises), the argument is referred to as a *valid* argument; otherwise it is referred to as an *invalid* argument. A valid argument is called *sound* if all of its premises are actually true. An *inductive argument* entails the claim that the conclusion follows with high probability, rather than necessarily, from its premises (e.g., one of the premises may take the form "Most Germans …" or "95% of all Germans …").[4] If the argument has a logical form that this claim is actually true (i.e., the conclusion does actually follow with high probability from the premises) the argument is referred to as a *strong* argument; otherwise it is referred to as a *weak* argument. A strong argument is called *cogent* if all of its premises are

[4] Note that the notion described here is just one kind of inductive inference, which is based on the application of an *inductive generalization*. Other inductive inferences are about inducing generalizations from examples. Inductive arguments can, but not need to, employ formal-statistical reasoning patterns.

actually true. Applying these definitions to real-world arguments is not straightforward. Some possible questions that may arise are: Which criteria to use to decide whether a premise is actually true or not? How to determine the level of probability of an inference if it is not explicitly expressed? And: What level of probability of an inference is good enough to classify an inductive argument as a strong argument?

Generally, the question arises who has the authority to judge the acceptability of arguments. For instance, as discussed in Van Eemeren and Grootendorst (2004, pp. 127–131), requiring that each premise itself needs further justification would lead into an infinite regress. So, the justificatory process has to stop at some point. Some criterion is in order to decide the most basic premises that are acceptable without further justification. One possibility here is to require that premises must be accepted by the addressed audience. Another is to require that dedicated representatives of the field in which the argumentation takes place decide (i.e., a panel, or forum, of experts).

1.2 Aristotelian Studies of Argumentation

One of the first systematic and comprehensive (Western) treatments of argumentation can be found in the work of Aristotle, which, to date, exerts a major influence on our thinking in philosophy and science. The following summary is based on an overview of Aristotle's work on argumentation provided by Van Eemeren et al. (1996).

In former times, the world was considered as predetermined. Natural events were controlled by the gods, so there was no need to ask for any further explanation. Things began to change in ancient Greece when people started to devise their own explanations of natural phenomena, for instance, asking questions of cause and effect (before, the will of a God was considered as the only acceptable cause of an event). Greek philosophers came up with different kinds of, partly contradictory, explanations to make sense of the world; different schools of thinking emerged. The question of which explanation or approach is the right one arose. Similarly, questions regarding the organization of the community came up, for instance, should everyone have an equal vote, or should the elite (i.e., the wisest and/or strongest) be in charge to steer the community? To systematically explore such issues and to convince others from one's own position, it was now essential to be a skilled arguer. Therefore, ancient Greeks became interested in the study of argumentation. A major contribution can be found in the work of Aristotle.

Aristotle distinguished two kinds of arguments: *deductive syllogisms* (the conclusion follows necessarily from the premises by means of logical entailment) and *inductive syllogisms* (the conclusion is derived from a number of specific cases by means of generalization). Furthermore, Aristotle classified arguments according to their purpose of use into three categories: demonstrative, dialectical, and rhetorical arguments. For each category he developed a theory: analytics, dialectic, and rhetoric, respectively. Table 1 summarizes the most relevant aspects regarding Aristotle's three categories of arguments.

Table 1

Argument categories distinguished by Aristotle: Demonstrative, dialectical, and rhetorical arguments

	Demonstrative arguments	**Dialectical arguments**	**Rhetorical arguments**
Objective	Certainty	Acceptability	Persuasiveness
Status of the premises	Evidently true	Acceptable	Persuasive to the audience
Inference	Valid	Valid	Persuasive to the audience
Theory	Analytics (logic)	Dialectic	Rhetoric

Note. Adapted from Van Eemeren et al. (1996, p. 33).

Analytics essentially refers to what is called nowadays *logic* and is concerned with demonstrative (or apodictic) arguments. The purpose of a demonstrative argument is to prove some conclusion with absolute certainty. The premises of a demonstrative argument are evidently true; the inference is logically valid. In his treatment of analytics, Aristotle focused on the examination of deductive syllogisms, and in particular, categorical syllogisms, that is, syllogisms concerned with the application of general categories. Table 2 shows the famous "Socrates" example.

Table 2

Socrates example of a deductive syllogism

(1) All humans are mortal. (*major premise*)

(2) Socrates is a human. (*minor premise*)

(3) Socrates is mortal. (*conclusion*)

Propositions (1) and (2) are the premises of the syllogism; proposition (3) is its conclusion. Each proposition is composed of a subject ["all humans" in (1), "Socrates" in (2) and (3)] and a predicate ["are / is mortal" in (1) and (3), "is a human" in (2)]. The *major premise* is the one that contains the predicate of the conclusion; the other premise, the *minor premise*, contains the subject of the conclusion. Major and minor premise share a *middle term* ["human(s)"] in the example), which bridges both premises to make the inference step possible. There are different syllogism variations. First, propositions may contain negation and a universal ("all") or existential ("some") qualifier. The resultant variations are called syllogism *moods*. Second, the middle term of the syllogism may be the predicate or the subject of minor and major premise. The resultant variations are called syllogism *figures*. All in all, Aristotle described 18 syllogism types based on different combinations of mood and figure. This systemization of syllogism types can be used as an analytic device to study deductive arguments in a uniform way.

Dialectic is concerned with dialectical arguments. Rather than rigorously demonstrating that some conclusion is true, dialectical arguments aim at defending one's standpoint and attacking the standpoint of an opponent in a debate situation. Analogously to demonstrative arguments, the inference step of a dialectical argument must be logically valid. Yet, the premises do not necessarily have to be evidently true but may just be generally accepted assumptions (i.e., "acceptable to all of the wise or to the majority or the most famous and distinguished from them"). The status of presented premises is often not immediately clear. Therefore, in the course of an argument, debaters try to bring the other party to make *concessions*, meaning that the opposing debater accepts premises presented by oneself. Aristotle describes general rules for conducting orderly debates and proposes a system of moves a discussant may undertake or parry in order to win the debate. Dialectics may also be exercised as a mental activity to systematically investigate the pros and cons of a given thesis. Overall, and in contrast to analytics, dialectic may be defined as "the art to argue for and against."

Rhetoric is concerned with rhetoric arguments. Aristotle's rhetoric studies investigate how to persuade an audience based on an orally held monologue. He distinguishes three relevant genres for rhetoric: before the court, in the political arena, and at ceremonial occasions. Arguers may employ means of three categories to be persuasive: *ethos* (practical wisdom, virtue, and good will), *pathos* (appeal to sentiment, e.g., joy, sorrow, love, hate), and *logos* (arguments). In his treatment of logos, Aristotle describes two kinds of rhetoric arguments: *enthymemes* (rhetorical deductive syllogisms) and *examples* (rhetorical inductive syllogisms). To be successful, the speech must take the specific type of audience into account. For instance, an arguer should rely on deductive reasoning to persuade a group of experts, while inductive, example-based reasoning may be more effective for an uneducated audience. Premises must be chosen to be acceptable by the audience, a looser standard as is used for analytic and dialectic arguments. Typically, premises are chosen to be at least plausible to the audience. In order to not annoy or waste the time of the audience, obvious premises may be omitted. In fact, the term *enthymeme* is often used in logic to specifically refer to an argument with missing premises or conclusion (Walton et al. 2006, p. 18).

1.3 Formal-logical Approaches to Argumentation

For a long time, formal logic was the method of choice to study arguments. The formal-logical approach requires that arguments are represented as expressions of a formal language, for instance, as sentences of propositional calculus. These sentences can be analyzed in terms of their logical properties, for instance, whether a given argument represents a deductively valid inference. To analyze natural language arguments using the formal-logical approach, one has to apply a number of abstractions to ultimately arrive at a formal expression. Table 3 demonstrates the process based on an example from van Eemeren et al. (1996).

The example demonstrates that formal logic deals with expressions that are reduced to their bare logical core. Many aspects that may be of interest to an argument analyst are stripped of from the original natural language statement or dialogue in the process of abstraction (e.g., linguistic, contextual, social, and psychological aspects are lost).

Besides this traditionally-oriented approach, other formal-logical approaches exist nowadays that are specifically tailored to model and analyze argumentative discourse. For instance, Barth and Krabbe's (1982) approach of *formal dialectics* yields a formalism to model a critical dialogue between a proponent and opponent in terms of general dialogue rules, an initial thesis, initial (and ongoing) concessions made by the discussants, attacking and defending moves, and conditions for winning the dialogue game. The presented calculus allows proving, with mathematical rigor, that an initial thesis can (or cannot) be established in light of a set of initial concessions made by an opponent (equivalent to the existence of a *winning strategy* of the proponent). The approach has been criticized to reduce dialogues to existing formal systems, rather than extending existing formal systems with pragmatic notions to better account for real-world dialogues (Stock 1982).

Table 3
Formal-logical analysis: From natural language argument to propositional argument form

Analysis step description	**Example analysis**	**Comment**
(1) Original transcript of conversation	Dale: Mary said she was going to get beef or cod. Do you know what we're eating tonight? Sally: No, but if she's already done the shopping it'll probably be in the fridge. I'll have a quick look in the fridge … It's stacked full. But I can't smell fish, anyway. Dale: O.K., as I see it, it is beef tonight since it was either that or cod and there is no fish. (*)	The statement to be analyzed is marked with an asterisk (*)
(2) Make reasoning explicit including all implicit elements	Dale: It was either beef or cod. There is no fish. If there is no fish, we are not going to eat cod. Therefore, it is beef tonight.	Makes the argument complete and self-contained
(3) Standardize representation by omitting references to persons, using uniform wording, and marking of premises and conclusion	*premise:* We are going to eat beef or we are going to eat cod. *premise:* There is no fish. *premise:* If there is no fish, we are not going to eat cod. *conclusion:* We are going to eat beef.	
(4) Replace natural language clauses with *sentence constants* while keeping words that define the sentence logic (*or*, *not*, *and*)	*premises:* B or C, not F, if not F then not C *conclusion:* B	B = We are going to eat beef C = We are going to eat cod F = There is fish
(5) Replace logic-related keywords with *logical constants*	$B \vee C$ $\neg F$ $\neg F \rightarrow \neg C$ $/\therefore B$	$\vee$ = or $\neg$ = not $X \rightarrow Y$ = if X then Y $/\therefore$ = therefore (conclusion)
(6) Determine whether a valid general *argument form* exists, which the specific argument is an *substitution instance* of	$p \cup q$ $\neg r$ $\neg r \rightarrow \neg q$ $/\therefore p$	Argument is valid since corresponding valid argument form exists

Note: Adapted from Van Eemeren et al. (1996, pp. 5–12).

Other more recent developments include the field of *computational dialectics* (Gordon 1996), which tries to formalize argumentative reasoning, e.g., to implement conflict-resolution behavior in multi-agent systems or support strategies in mediation systems for online discussions. One particular class of logical systems, *non-monotonic logics*, has been identified as particularly relevant to model argumentation (Gordon et al. 2007). In classical, monotonic logic, the set of sentences entailed by a theory (or knowledge base) does monotonically increase when new axioms are added to the theory. Adding axioms that do not contradict the theory only allows inferring new sentences; adding axioms that contradict the theory makes the theory inconsistent, meaning that any sentence can be inferred. Non-monotonic logic systems, on the other hand, allow *defeasible inferences*. Adding new sentences to a theory can defeat existing inferences, that is, previously, *prima facie*, justified sentences may be withdrawn again. Similarly, argumentation may be considered as a non-monotonic process since statements and inferences accepted at one point can be withdrawn again when new arguments or evidence are presented. Gordon et al. (2007) criticize the static nature of non-monotonic models of argumentation and propose their own formal model, *Carneades*, which takes procedural aspects into account. In particular, Carneades allows modeling, on a statement-by-statement basis, the current *dialectical status* (i.e., statement is stated, questioned, accepted, or rejected), the allocation of the *burden of proof* (i.e., which party is obliged to substantiate a statement), and the *proof standard* that applies (i.e., the rules that decide whether a statement is acceptable or not; e.g., “scintilla of evidence,” or “best argument”). This information can be employed to model important procedural aspects, such as the overall procedural context (e.g., in criminal trials, a defendant’s guilty must be proven with evidence “beyond reasonable doubt”), the current procedural stage (e.g., at the beginning of a deliberation dialogue, when brainstorming ideas, one may apply a low standard of proof), changing dialectical statuses (e.g., conceding to a statement of the other party alters its dialectical status to “accepted”), and changing obligations of involved parties (e.g., questioning an implicit assumption of an argument typically shifts the burden of proof back to the originator of the argument).

1.4 Toulmin: The Uses of Argument

A very influential and highly cited treatment of argumentation is the book *The Uses of Argument* by the British philosopher Stephen Toulmin (2003; original version published in 1958). Toulmin criticized that the at that time prevalent approaches to study argumentation using rigid deductive methods of formal logic were not suitable to analyze and judge significant real-world arguments. Strictly logical deductions, based on the concept of logical necessity, are only "an unrepresentative and misleadingly simple sort of argument" (p. 135). With his criticism, Toulmin had a major influence on current approaches to argumentation, such as *informal logic*, which focuses specifically on real world arguments and uses a broader conception of argumentation. In particular, informal logic understands argumentation as inherently *social* (argumentation as a social practice), *dialectical* (argumentation as an actual or anticipated dialogue to resolve a conflict of opinion in a rational way), and *pragmatic* (argumentation as an event in a meaningful context) (Van Eemeren et al. 1996, p. 164). Most arguments in informal logic are not deductively valid but *defeasible*, meaning that their conclusions are only provisionally accepted until doubts are raised or new negative evidence comes in. (Note that newer formal-logical approaches can partly accommodate for the shortcomings criticized by Toulmin, see the discussion of more recent formal approaches above.)

A second important contribution of Toulmin is the notion of *field dependence* of argumentation. He observed that, depending on the specific type of assertion made, people use different kinds of reasons and different standards for assessing the reasons brought forward by others. For instance, arguments may take quite different forms depending on whether a legal decision, a moral judgment, a scientific theory, a mathematical theorem, or an assertion about the aesthetic qualities of a piece of art is justified. Accordingly, arguments can be categorized into different logical types depending on the nature of conclusion and data. Toulmin used the term *fields of argument* to distinguish such different logical types. An important question, according Toulmin, is which aspects of arguments are *field-invariant* and which ones are *field-dependent*. While criteria to judge the quality and sufficiency of an argument essentially depend on the specific field of argument, there is a baseline set of components that may be considered in the analysis of any rational argument, across different fields. Toulmin proposed a specific layout, later referred to as the Toulmin model, which encompasses these components. The Toulmin model of argument has been widely adopted by scholars across different academic branches, including education, and is still cited in much research and analysis of argumentation today.

1.4.1 Toulmin model of argument

Toulmin considered the classical Aristotelian syllogism “minor premiss, major premiss, so conclusion” as overly simplistic and misleading and therefore not suitable to represent and analyze the structure of arguments. Based on reflections on a typical legal procedure, he proposed a richer model of argument, claiming that an analysis of the logical process more generally requires "a pattern of argument no less sophisticated than is required in the law" (p. 89). The Toulmin model distinguishes the following components:

- *Claim:* A proposition or conclusion one wants to establish.
- *Data:* The facts or information one brings forward in order to establish the claim. Essentially, what one would answer when asked "Why?" in response to one's claim.
- *Warrant:* Statements, typically of a more general nature, that establish the step from data to the claim. Essentially, what one would answer when asked "How do you get there?" in response to the presented claim and data. The warrant determines how relevant the data is with respect to the claim.
- *Qualifier:* Modal qualifiers (e.g., *necessarily*, *probably*, and *presumably*) used to express the force of the warrant to authorize the inference from data to claim.
- *Rebuttal:* Exceptional conditions under which the warrant does not legitimate the inference from data to claim.
- *Backing:* Assurance why the general rule expressed in the warrant is legitimate. The nature of the backing typically depends on the specific field of argument (e.g., a legal claim may be backed by a legal statute; a claim regarding the genus of some animal may be backed by a system of taxonomic classifications).

Figure 3 shows the Toulmin model and its application to an example argument. An important limitation of the Toulmin model is its focus on the structure of a single argument. The voice of a possible opponent is not explicitly represented in the model. So the dialectical aspect of argumentation, the weighing of pro and con arguments to make a decision about the acceptability of a claim, is not reflected in the model (Van Eemeren and Grootendorst 2004, p. 47).

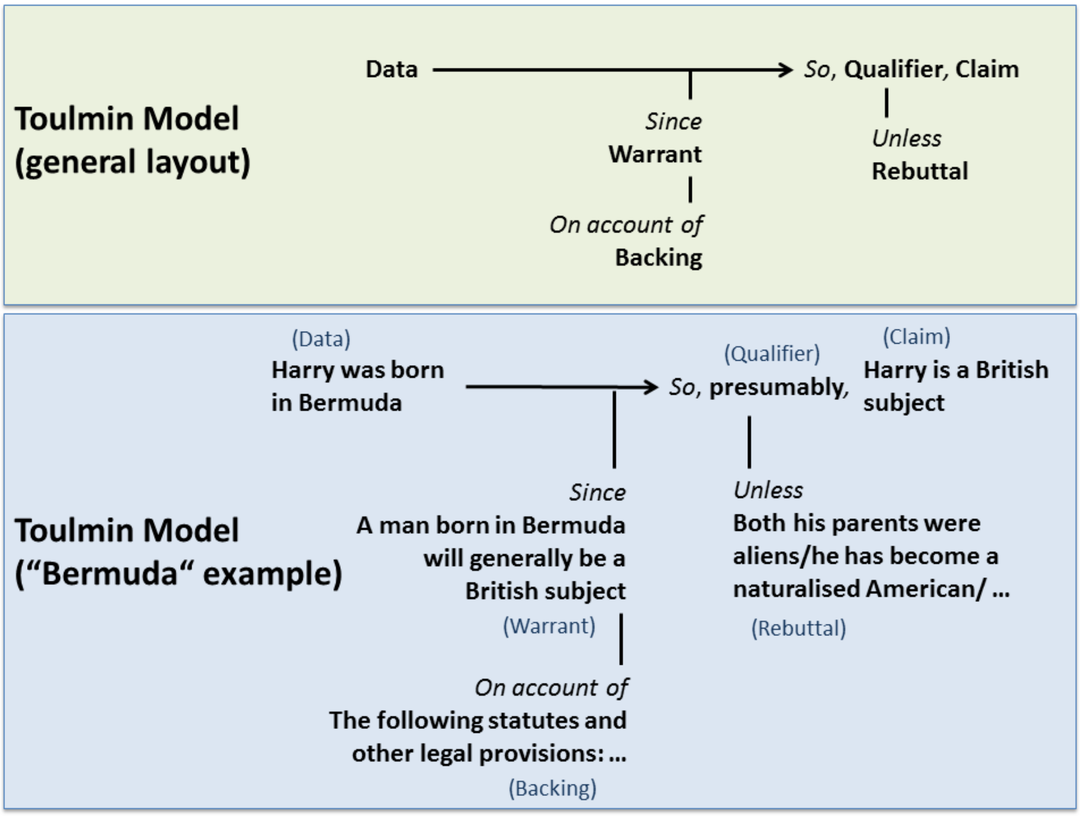

Toulmin Model
(general layout)
Data
So, Qualifier, Claim
Since
Warrant
On account of
Backing
Unless
Rebuttal
Toulmin Model
("Bermuda" example)
(Data)
Harry was born in Bermuda
So, presumably,
(Qualifier)
(Claim)
Harry is a British subject
Since
A man born in Bermuda will generally be a British subject
(Warrant)
On account of
The following statutes and other legal provisions: …
(Backing)
Unless
Both his parents were aliens/he has become a naturalised American/ …
(Rebuttal)

Figure 3: General layout of Toulmin model (upper part) and Bermuda example (lower part). Adapted from Toulmin (2003, p. 97).

1.5 Philosophical and Linguistic Theories of Language Use in Dialogue

As discussed, real-world argumentation may considerably differ from what formal logical approaches are currently able to represent and model. Therefore, argumentation theorists have opened up to other fields and perspectives that may enhance the understanding of practical argumentation. Since argumentation is primarily a phenomenon of natural language, research in the fields of philosophy of language and linguistics had a major influence on argumentation theory.

To analyze natural language expressions, one may look at different aspects of language. On a shallow level, one may be interested in how individual words can be combined in order to yield well-formed sentences. A set of rules, or *grammar*, may be employed to systematically describe the regularities that determine which combinations are permissible in a given language and which ones not. The field of study concerned with such kinds of analysis is referred to as *syntax*. Languages are sign systems; signs stand for something other than themselves; signs have a meaning.

So, on a deeper level, one may ask about the meaning of words and sentences. The field of study concerned with such kinds of analysis is referred to as *semantics*. For instance, the formal-logical analysis presented in section 1.3 can be considered as a semantic analysis. Finally, to get a proper understanding of real world communication, it is often not sufficient to confine the analysis to the literal meaning of language expressions. It requires knowledge of the broader context in which communication takes place. Table 4 illustrates this point using an example from Levinson (1983; pp. 97–98). As can be seen, the literal meaning of sentences often largely departs from human understanding of language expressions.

Table 4
Formal-logical analysis of dialogue contrasted with human understanding

Level of interpretation	**Example**
(1) Original transcript of conversation	A: Can you tell me the time? B: Well, the milkman has come
(2) Interpretation on a purely semantic level, what is literarily expressed	A: Do you have the ability to tell me the time? B: [particle] the milkman came at some time prior to the time of speaking
(3) Likely human-understanding of conversation moves, what is communicated (materials in *italics* are likely to be implicitly added by human inference)	A: Do you have the ability to tell me the time *of the present moment, as standardly indicated on a watch, and if so please do so tell me* B: *No, I don't know the exact time of the present moment, but I can provide some information from which you may be able to deduce the approximate time, namely* the milkman came at some time prior to the time of speaking

Note: Example from Levinson (1983, pp. 97–98).

Other examples include the use of irony and metaphors, which can only be understood when going beyond the literal meaning of words and sentences. The province of such analyses is the field of *pragmatics*. Linguistic pragmatics is concerned with the meaning of natural language expressions in context, beyond what is literary said or written.[5] To emphasize the context aspect, the term *utterance* may

[5] The borderline between semantics and pragmatics is not as clear as it may initially seem. The question is how the "literal" or semantic meaning of language expressions should be defined. A semantic interpretation also involves background knowledge not contained in the natural language expressions themselves, e.g., to map, in a consistent way, sentences and sentence fragments to logical propositions. Pragmatic aspects may also be accounted for by semantic theories, although this may

be employed with the meaning “sentence in context.” Levinson (1983) discusses several focal areas of study in pragmatics.

- The study of *conversational implicatures* is about conveyed but not explicitly expressed aspects of message content. The omission of such aspects in utterances is based on the sender’s assumption that the receiver will implicitly infer those aspects. The conversation in Table 4 is an example for conversational implicature.
- The study of *deixis* is concerned with grammaticalized references to the context of speaking. For instance, the word *now* may be used as a reference to a point in time, the word *you* as a reference to a person, and the word *here* as a reference to a place. Sentences containing such expressions cannot be fully understood out of context (what point in time? what person? what place?).
- The study of *presuppositions* is about implicit assumptions required for the meaningful interpretation of utterances. For instance, the utterance “the King of France is wise” presupposes the existence of a King of France.
- The study of *speech acts* holds the perspective that linguistic expressions used in practice must be conceived of more broadly, as intentional actions, rather than mere declarative, truth-bearing sentences.

This section focuses on two aspects: the *principle of cooperativeness*, which explains the role of conversational implicatures in discourse, and the *theory of speech acts*. Both theories are foundational for the pragma-dialectical theory of argumentation discussed below. Moreover, this section discusses the theory of grounding in communication, which is a process considered pivotal to the success or the failure of discourse (including argumentation).

1.5.1 Principle of cooperativeness

As illustrated in Table 4, the proper understanding of language requires inferences that cannot be made on the sole basis of the propositional content (or literal meaning) of uttered sentences. Likewise, speakers seem to know which inferences they can expect their hearers to make. Grice (1975) proposed that a general principle of cooperativeness, which can be subdivided into more specific behavioral maxims, governs the production and interpretation of language. In particular, the overarching cooperative principle is:

blow up the complexity and reduce the conceptual coherence of the theory. The issue is discussed in detail in Levinson (1983, pp. 1–35).

> "Make your conversation contributions such as required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged."

The more specific maxims are:

- *Maxim of quantity*
  1. "Make your contribution as informative as required (for the current purpose of the exchange)."
  2. "Do not make your contribution more informative than is required."
- *Maxim of quality:* "Try to make your contribution one that is true."
  1. "Do not say what you believe to be false."
  2. "Do not say that for which you lack adequate evidence."
- *Maxim of relevance:* "Be relevant."
- *Maxim of manner:* "Be perspicuous."
  1. "Avoid obscurity of expression."
  2. "Avoid ambiguity."
  3. "Be brief (avoid unnecessary prolixity)."
  4. "Be orderly."

The hearer is able to make inferences of the kind discussed above based on the assumption that the speaker is trying to be cooperative and hence complying with the proposed maxims. To draw a distinction between semantic inferences based on the utterance's propositional content and inferences based on the cooperative principle, Grice introduced the term *conversational implicatures* for the latter. To illustrate the four maxims and the kinds of inference they trigger, Table 5 gives an example of each.

Table 5
Pragmatic inferences explained through Grice's cooperation maxims

What is said	What is additionally communicated	Maxim at work and explanation
Nigel has 14 children.	Nigel has not more than 14 children.	Maxim of quantity If Nigel had more than 14 children, the statement would violate the maxim of quantity since not the full set of required information is provided
Does your farm contain 400 acres?	I don't know that it does, and I want to know if it does.	Maxim of quality For a question to be sincere, the expectation is that one is actually lacking and requiring the information one asks for.
A: Where is Bill? B: There's a yellow VW outside Sue's house.	Bill may be in Sue's house	Maxim of relevance On face value, the reply seems to be irrelevant. The hearer assumes cooperativeness and thus searches for an inference that makes the reply relevant to the question. Since Bill owns a yellow VW, the suggested inference is a reasonable interpretation.
Walk up to the door, turn the door's handle clockwise as far as it will go, and then pull gently towards you.	It is important to exactly and carefully follow this advice to open the door.	Maxim of manner If details were not important, the speaker could just say "Open the door." Otherwise, the speaker would violate the sub-maxim to be brief.

Note: Examples taken from Levinson (1983, pp. 100–108).

The speaker may also intentionally flout cooperation maxims in order to convey a nonliteral meaning. Our ability to understand certain figures of speech, such as irony, may be explained based on this hypothesis. For instance, consider the conversation in Table 6 (taken from Levinson 1983).

Table 6
Example of the use of irony

A: What if the USSR blockades the Gulf and all the oil?
B: Oh come now, Britain rules the seas!

Note: Example taken from Levinson (1983, p. 109).

Since it is generally known that Britain does not rule the seas, *A* is invited to come up with an alternative interpretation of *B*'s utterance, one that is compliant with the cooperation principle. An obvious interpretation is the use of irony, that is, *B* may actually mean exactly the opposite of what he said (i.e., "Britain does not rule the seas”). So, by making a further inference, *A* may interpret *B*’s answer as the judgment that there is nothing Britain can do about a possible blockade of the Gulf. *B* can safely use irony because he knows that *A* assumes that he is following the cooperation principle and therefore will be able to make the right non-conventional inference.

1.5.2 Speech act theory

Speech act theory was originally proposed by Austin (1962) in his book *How to Do Things with Words*, and further developed by Searle (1969). This section draws from an overview given by Levinson (1983).

Speech act theory is based on the observation that utterances are not only representations of propositional content, which may be verified in terms of some semantic theory, but can also be seen as intentional actions. Austin (1962) started his treatment based on a distinction between *constatives*, that is, utterances that make statements about the world, and *performatives*, that is, utterances used to perform certain actions with potential bearing on the state of the world (e.g., “I sentence you to ten years of hard labor” or “I declare war on Zanzibar,” Levinson 1983). Since the performative aspect cannot be analyzed with the classical approach of *truth conditions*, Austin introduced the notion of *felicity conditions*, which describe conditions a performative must satisfy in order to be successful (or “felicitous”). For instance, to deliver a verdict or to declare war, the speaker must have the authority to do so. In the course of his argument, Austin (1962) developed the concept further to arrive at a radically extended notion of speech acts. Rather than being a property of one specific kind of utterance (i.e., performatives), performative aspects may be found in essentially any utterance. For instance, making a request, an assertion, or a promise are in fact intentional actions (which may or may not employ some propositional content).

To give a more detailed account of the nature of these actions, Austin proposes that each utterance goes with three different simultaneously performed acts that exert specific forces (Levinson 1983, p. 236):

- a *locutionary act*, which involves the “utterance of a sentence with determinate sense and reference”
- an *illocutionary act*, which involves the “making of a statement, offer, promise, etc.” Essentially, the illocutionary act corresponds to the conversational

intention of the utterance (i.e., what the speaker wants to achieve). Often, the term *speech act* is used to specifically refer to the illocutionary act.

- a *perlocutionary act*, which is about the actual effect the utterance has on the audience. The effect might be intended or unintended, and typically depends on the specific circumstances.

Searle (1969, 1976) extended and refined Austin's theory. Among other things, he proposed five basic types of speech acts, which could be used to classify utterances (Levinson 1983, p. 240):

- *representatives* (or *assertives*) are speech acts that commit the speaker to the truth of the expressed proposition (e.g., asserting, concluding)
- *directives* are speech acts that try to get the addressee to do something (e.g., requesting, questioning)
- *commissives* are speech acts that commit the speaker to some future course of action (e.g., promising, threatening, offering)
- *expressives* are speech acts that express a psychological state (e.g., thanking, apologizing, welcoming, congratulating)
- *declaratives* are speech acts with immediate effect. They typically rely on specific institutional contexts and procedures. They are essentially the performatives described by Austin. (e.g., excommunicating, declaring war, christening, firing from employment)

The concept of speech acts has been adopted in many areas, including argumentation theory and computational approaches to modeling and supporting educational discussions. Searle's typology of speech acts is just one approach. Many other typologies, both general and specific to certain fields of application, have been developed since then.

1.5.3 Organization of discourse

This section extends the perspective from single utterances to the organization of utterances in discourse. Discourse, as a complex collective activity, requires considerable efforts in producing contributions in an orderly and coordinated way. This section discusses the concepts of *turn-taking*, *adjacency pairs* (both based on Levinson 1983), and *grounding* (Clark and Schaefer 1989), which is a process essential for discussants to maintain a common understanding of the current state of the discussion.

Multiparty discourse requires the transition of the speaker role between the different participants. The ease with which humans manage to accomplish this is astonishing. Despite the fact that the gap between two consecutive utterances is often not more than a few micro-seconds, humans are incredibly successful in avoiding overlaps in speaking. The talking of an individual speaker until the next speaker starts is referred to as a *turn*. The process governing the decision when who is to speak next is referred to as *turn-taking*. Sacks et al. (1978) propose that turn-taking is based on a local management system comprising a set of rules that manage, on a turn-by-turn basis, control over a scarce resource (the speaker role) in an economic way (minimizing gap and overlap). The system is based on cues for requesting or releasing the *floor* (i.e., the speaker role). Turns are conceived of as a series of *turn-constructional units*, which are identified based on syntactic, prosodic, and intonational properties. Turn-constructional units mark places where the speaker may change (called *transition-relevance places*). The next speaker may be selected by the current speaker through specific means, such as gaze or a direct address by name. Sacks et al. (1974) propose the following rules that govern the transition of speaker turns (simplified, taken from Levinson 1983, p. 298, *C* = current speaker, *N* = next speaker):

1. "If *C* selects *N* in current turn, then *C* must stop speaking, and *N* must speak next; transition occurring at the first transition-relevance place after *N*'s selection"
2. "If *C* does not select *N*, then any (other) party may self-select, first speaker gaining right to the next turn"
3. "If *C* has not selected *N*, and no other party self-selects under option (2), then *C* may (but need not) continue (i.e. claim rights to a further turn-constructional unit)"

A typical pattern reflecting an important organization principle in dialogues is the occurrence of *adjacency pairs* (Schegloff and Sacks 1973). An adjacency pair consists of two parts (first and second part) that correspond to utterances of different speakers that relate to one another (part two is essentially a response to part one). Examples for adjacency pairs are question-answer, greeting-greeting, and offer-acceptance pairs. The two parts may occur next to each other (which was in first definitions one of the required properties, therefore *adjacency* pairs), but may also be separated by intermediate utterances. For instance, a question may not be properly understood and therefore responded to by a clarification question rather than a direct answer. So some sub-dialogue, or *insertion sequence*, may separate the first and the second part of the adjacency pair. Therefore, the criterion of strict adjacency is relaxed to a criterion of *conditional relevance*, which only prescribes the existence of an identifiable second part that is relevant and expectable with respect to the first part. Besides the prototypical second part one would expect in the first place based on the first part (e.g., an "answer" in response to a "question"), the second part may also be of a different category. For instance, discussants may reject to answer a question when they are dubious whether the question is meant sincerely. Therefore, a distinction is made between *preferred* and *dispreferred* responses. Dispreferred responses are typically *marked* in discourse, e.g., a refusal to answer a question may be preceded by a delay or filler words (e.g., *uhm*).

A concept conducive to the understanding of discourse is *common ground*, which refers to the knowledge shared between participants of a discussion. The kind of shared knowledge is not restricted to factual knowledge. For instance, the principle of cooperation, discussed in section 1.5.1, can be considered as part of the common ground, likewise discourse norms in general and in specific institutional settings (e.g., in the courtroom), shared beliefs, and shared assumptions. Based on the presupposition that other discussants already possess—or can easily infer—certain knowledge, a speaker may be less than fully explicit in his communication. Discourse can be understood as a process in which participants continuously try to add new items to the common ground. In this way, the common ground accumulates over the course of a discussion. For instance, to add a piece of factual knowledge, one may explain some fact. To add a belief about a controversial standpoint, one may present supporting arguments to establish that belief. The common ground may also include knowledge about the current state of the discussion. For instance, when a person asks a question, a proposition that this person wants to know something may be added as well. If the question is answered, the proposition may be *destroyed* again and replaced by a new proposition that the person now knows the answer to her question.

Clark and Schaefer (1989) propose a process called *grounding*, which ensures that the common ground can accumulate in an orderly way. In the process of grounding, discussants try to establish the mutual belief that what is said is also understood, and hence part of the common ground. Contributing to discourse therefore involves two interwoven processes:

1. *Content specification:* The speaker tries to specify some propositions and the listeners try to register that proposition.
2. *Grounding:* Speaker and listeners try to achieve the *grounding criterion*, which is:
   - "The contributor and the partners mutually believe that partners have understood what the contributor meant to a criterion sufficient for current purposes."

A contribution to discourse may thus be conceived of as a collective act consisting of a speaker presenting an utterance (*presentation phase*), and a listener giving evidence of his understanding of the utterance (*acceptance phase*). This evidence may take different forms, which constitute the strength of evidence. Weak kinds of evidence are to just keep attending to the speaker or to continue with a new contribution without signaling a lack of understanding. Stronger kinds of evidence include an explicit acknowledgment or some form of displaying or demonstrating understanding. If a speaker realizes that the listeners' understanding is not sufficient, he will try to *repair* the problem in understanding. A full understanding is often not required, so, for economic reasons, repairs are only initiated when the speaker considers the listeners' understanding as not "sufficient for current purposes." Clark and Schaefer (1989) describe four levels of understanding (listed here from weakest to strongest):

1. not noticing that speaker uttered something
2. noticing that speaker uttered something
3. correctly hearing the content of the utterance
4. understanding the meaning of the utterance

Repairs are initiated as soon as possible; otherwise problems may start to "snowball." This is in line with the *least collaborative effort* principle, which says that discussants organize their discourse in a way that minimizes the total effort involved in a contribution (a snowballing of problems would be considerably more expensive in terms of effort than a prompt repair). Table 7 illustrates the process of grounding.

Table 7
Example of grounding in communication

Transcript of conversation	**Interpretation according to theory of grounding**
A: is it . how much got Norman get off --	Presentation of utterance with initial self-correction (immediate self-corrections are the most cost-effective way to address problems)
B: pardon	Signaling trouble in understanding (at the level of not fully hearing the content of the contribution)
A: how much does Norman get off	Repairing problem in understanding (by re-presenting question)
B: oh, only Friday and Monday	Signaling awareness of repair move ("oh") and understanding (implicitly, by giving a reasonable answer)
A: m	Accepting response through *continuer* ("m")
B: *[continues]*	…

Note: Example taken from Clark and Schaefer (1989).

1.6 Van Eemeren and Grootendorst: The Pragma-dialectical Theory

One of the most significant current theories of argumentation is Van Eemeren and Grootendorst's (2004) pragma-dialectical theory, which is based on an integration of the logico-centric, dialectical tradition of studying argumentation with insights from linguistic pragmatics. Van Eemeren and Grootendorst argue that neither of these approaches alone can sufficiently account for real-world argumentation in its entirety. The *dialectical* view can contribute a normative, abstract model of how argumentation should look like from the perspective of rationality. The *pragmatic* view can contribute a descriptive account of how argumentation unfolds under real-world constraints. In line with this *normative-pragmatic* view, Van Eemeren and Grootendorst propose the following definition of argumentation (p. 1):

> "Argumentation is a verbal, social, and rational activity aimed at convincing a reasonable critic of the acceptability of a standpoint by putting forward a constellation of propositions justifying or refuting the proposition expressed in the standpoint."

The dialectical aspect is reflected by defining argumentation as "rational" and "aimed at convincing a reasonable critic." The pragmatic aspect is reflected by defining argumentation as an “activity” that is “social” in nature. Individual argumentative reasoning is conceived of as a discussion with an imagined opponent.

Table 8

Phases of a critical discussion and corresponding speech acts according to the pragma-dialectical theory

Phase	**Description**	**Relevant speech act types**	**Speech act usage**
Confrontation	Conflict of opinion emerges	Assertives	Express a standpoint
		Commissives	Acceptance or non-acceptance of standpoint
			Upholding non-acceptance of standpoint
Opening	Clarification of common ground (background knowledge, values, expectations with respect to discussion format and roles)	Directives	Challenging to defend a standpoint
		Commissives	Acceptance of the challenge to defend a standpoint
	Decision whether or not to start the discussion (e.g., discussion may not be started if expectations regarding discussion format largely diverge)		Agreement on premises and discussion rules
			Decision to start discussion
	Opening stage often stays implicit since existence of sufficient common ground is typically tacitly assumed by discussants		
Argumentation	Core argumentation activities (challenging, arguing, counter-arguing, conceding, etc.)	Directives	Requesting argumentation
		Assertives	Advancing argumentation
		Commissives	Acceptance or non-acceptance of argumentation
Concluding	Drawing the conclusion	Commissives	Acceptance or non-acceptance of standpoint
		Assertives	Upholding or retracting a standpoint
			Establishing the result of the discussion

Note: Adapted from Van Eemeren and Grootendorst (2004, p. 68).

On a more detailed level, Van Eemeren and Grootendorst (2004) propose an ideal model of a *critical discussion* that unites elements from the dialectical and the pragmatic view. The model serves a *heuristic function* (as a tool to analyze and interpret argumentation) and a *critical function* (as a standard for evaluating the quality of argumentation). As a prerequisite for a critical discussion, a conflict of opinions is assumed. A protagonist puts forward a thesis and tries to convince a critical antagonist of the acceptability of that thesis. In the case of a monologue, the antagonist is not real but imagined or anticipated. The critical discussion ends with either the antagonist being convinced, or the protagonist withdrawing the thesis. Four phases of the critical discussion are distinguished each corresponding to a set of dialogue acts appropriate in that phase (see Table 8). The typology of speech acts is based on the scheme proposed by Searle (1976) discussed above. In addition, Van Eemeren and Grootendorst propose to define one specific, particularly relevant subclass of declaratives: *usage declaratives*. In contrast to the declaratives defined by Searle (1976), usage declaratives do not depend on any specific institutional context. Their aim is to clarify the meaning of other speech acts (e.g., by explaining or defining content elements of previous speech acts). Usage declaratives (e.g., giving some explanation) and directives to request usage declaratives (e.g., asking for an explanation) can be used in all phases and are not included in Table 8. Based on Grice's cooperative principle discussed above, Van Eemeren and Grootendorst (2004) propose criteria for the execution of speech acts (e.g., speech acts must be comprehensible).

Protagonist and antagonist of an ideal critical discussion behave according to specific norms. Van Eemeren and Grootendorst (2004) describe 15 discussion rules that specify a procedure in accordance with these discussion norms. They argue that their rules satisfy two conditions that are central for the reasonableness of the rules. First, the rules must be effective in guiding discussants towards the resolution of their conflict of opinion (*problem validity*). The rules should not only commit discussants to the making of proper (logical) inferences, but also lead them towards accomplishing the specific (pragmatic) tasks of the different stages described in Table 8. Second, the rules must take a form such that they are acceptable to protagonist and antagonist (*conventional validity*). If protagonist and antagonist are honestly interested in resolving their conflict of opinion in a rational way, they are likely to accept a set of rules that is problem-valid. Since the rules are of a rather technical nature and thus not very accessible to "real" discussants, Van Eemeren and Grootendorst (2004) propose a more straightforward and better understandable "code of conduct":

1. “Discussants may not prevent each other from advancing standpoints or from calling standpoints into question.”
2. “Discussants who advance a standpoint may not refuse to defend this standpoint when requested to do so.”
3. “Attacks on standpoints may not bear on a standpoint that has not actually put forward by the other party.”
4. “Standpoints may not be defined by non-argumentation or argumentation that is not relevant to the standpoint.”
5. “Discussants may not falsely attribute unexpressed premises to the other party, nor disown responsibility for their own unexpressed premises.”
6. “Discussants may not falsely present something as an accepted starting point or falsely deny that something is an accepted starting point.”
7. “Reasoning that in an argumentation is presented as formally conclusive may not be invalid in a logic sense.”
8. “Standpoints may not be regarded as conclusively defended by argumentation that is not presented as based on formally conclusive reasoning if the defense does not take place by means of appropriate argument schemes that are applied correctly.” *(Essentially, a non-deductive argumentation must employ an appropriate and correctly applied argumentation scheme.)*
9. “Inconclusive defenses of standpoints may not lead to maintaining these standpoints, and conclusive defenses of standpoints may not lead to maintaining expressions of doubt concerning these standpoints.”
10. “Discussants may not use any formulations that are insufficiently clear or confusingly ambiguous, and they may not deliberately misinterpret the other party’s formulations.”

1.7 Walton: Argumentation Schemes, Critical Questions, and Fallacies

The critical discussion described by Van Eemeren and Grootendorst (2004) is only one possible type of discussion. Walton (2008) describes six main types of dialogue in which argumentation plays a role (see Table 9). At the opening stage of a discussion, participants agree, implicitly or explicitly, on the type of discourse they will engage in. It is possible that the type of discourse shifts in the course of the discussion (e.g., in the worst case, a rational and objective discussion may drift towards a personal quarrel).

Walton's description of a *critical discussion* (or *persuasion dialogue*) largely corresponds with Van Eemeren and Grootendorst's (2004) model. That is, starting point is a conflict of opinion, participants behave cooperatively and rationally, following specific rules and norms, and the discussion concludes when the issue is clarified. Walton makes a distinction between two subtypes: On the one hand, a *dissent* is characterized by an asymmetric situation in which a proponent wants to persuade a skeptical opponent from his standpoint. The *burden of proof* is on the side of the proponent, who has to show that his standpoint is acceptable in light of the concessions made by the opponent. The opponent has a lesser obligation since he only has to express reasonable critique against the standpoint. On the other hand, a *dispute* is characterized by a symmetric situation in which both parties equally have the burden of proof with respect to their (opposing) standpoints.

Table 9
Walton's typology of dialogues

Type of dialogue	**Initial situation**	**Participant's goal**	**Goal of dialogue**
Persuasion (Critical discussion)	Conflict of opinion	Persuade other party	Resolve or clarify issue
Inquiry	Need to have a proof	Find and verify evidence	Prove (disprove) hypothesis
Negotiation	Conflict of interest	Get what you most want	Reasonable settlement both can live with
Information-seeking	Need to have information	Acquire or give information	Exchange of information
Deliberation	Dilemma or practical choice	Co-ordinate goals and actions	Decide best available course of action
Eristic (Quarrel)	Personal conflict	Verbally hit out the opponent	Reveal deeper basis of conflict

Note: Adapted from Walton (2008, p. 8)

In stark contrast to the critical discussion stands the *quarrel* (or *eristic* discussion). The quarrel constitutes the lowest level of argumentation. Rather than aiming at resolving a difference of opinion in a rational and objective way, participants try to verbally attack their opponent on a personal level. They do not feel committed to standards of reasonableness and fairness but rather try to win the conflict at any cost. Heated emotions often play an important role in quarrels.

The *debate* may be seen a mixture between a critical discussion and a quarrel. On the one hand, a debate is often regulated through a set of rules that ensure that the dialogue takes an orderly form and hinders the excesses characteristic for quarrels. On the other hand, the rules are often very permissive, allowing many forms of fallacious argumentation. The participants' sole focus is on winning the debate, often at the expensive of standards of sound reasoning. This typically leads to the use of rhetoric tricks and appeals to emotion to impressive the audience or disconcert the opponent. Nevertheless, debates take an important function in society, for instance, in the political arena.

Other main types of dialogue include the *negotiation* (balancing out personal interests rather than establishing truth), the *inquiry* (inferring reliable knowledge based on a careful scrutiny of known facts rather than concessions made by the discussants, e.g. in science), the *deliberation* (making of well-reasoned practical choices between alternative actions), and the *information-seeking dialogue* (exchanging information).

An important contribution of Walton is his extensive work with respect to *argumentation schemes* and *fallacies*. Research on these two important aspects can be traced back to Aristotle. Argumentation schemes describe general forms of inference used to persuade others of one's standpoint. Most argumentation schemes deviate from the traditional, logico-centric ideal of a deductively valid argument. Rather, any kind of inference aimed at transporting the acceptability of a set of premises to a conclusion may be represented by means of an argumentation scheme, including arguments that are defeasible (a detailed explanation follows below). Similarly, modern argumentation theory declines the view that any deductively invalid argument must be considered as a fallacy. Instead, arguments are judged in terms of their appropriateness with respect to the goals of the dialogue. Therefore, the decision whether an argumentation scheme is applied in an appropriate or fallacious way strongly depends on the specific type of dialogue and context, and may require a careful case-by-case analysis.

The notion of *defeasibility* is of central importance to current argument-theoretical approaches. Once the premises of a deductive argument have been accepted, an obligation arises to also accept its conclusion. In contrast, the acceptance of the premises of a defeasible argument only leads to a provisional acceptance of its conclusion. The logical structure of a defeasible inference leaves room for exceptional cases in which the conclusion does not necessarily follow from its premises. So, if new information enters the picture, or serious doubts are raised, the conclusion may not be accepted anymore even if its premises have been accepted. The defeasible nature of arguments can often be represented through a *generalization premise* that takes the form of a *defeasible*, or *non-strict*, generalization (such as a defeasible modus ponens, e.g., "Generally, if X then Y." or "If X then plausibly Y."). Generalization premises correspond to warrants in the Toulmin model and are typically not explicitly mentioned in real-world discourse. Often, the acceptability of defeasible arguments also hinges on other elements not explicitly addressed in the argument. For instance, an *argument from expert opinion* (e.g., "Expert X said Y.") contains a number of implicit presumptions that may be questioned, for instance: Is X really an expert in a field that allows him to give a competent judgment about Y? Or: Is X honest about Y? Therefore, such arguments are also called *presumptive arguments*. Presumptions may be encoded as additional premises of the argumentation scheme. Walton et al. (2008, pp. 15–21), however, favor to represent presumptions in the form of *critical questions* and to reserve premises for the core logic of argumentation schemes. The critical questions can serve as a handy tool for analysts or trained discussants to probe whether the argumentation scheme has been applied in an appropriate or fallacious way. When used in a discussion, critical questions shift the burden of proof back to the presenter of the argument. That is, before the argument is accepted as such, the presenter must clarify and substantiate the implicit presumptions that were brought to the foreground through critical questions. Walton et al. (2008, p. 10) explicitly draw a line between deductive and inductive argumentation schemes on the one hand and defeasible ones on the other. Valid deductive and strong inductive arguments have a more stringent internal logic and are therefore more binding than defeasible arguments. However, due to uncertainty and lack of knowledge, people often have to resort to weaker forms of inference.

In summary, argumentation schemes semi-formally describe prototypical forms of inference in terms of premises, a conclusion, and critical questions. In particular, defeasible argumentation schemes are of central importance for the study of real-world discourse from an argumentation-theoretical perspective. Argumentation schemes may not only be used by analysts to help identify implicit premises and evaluate argumentation. They may also be employed in education, to impart the skills of proper argumentation, and in artificial intelligence research, as a blueprint to build computational models of argumentation.

Walton et al. (2008) present a compendium of 65 argumentation schemes based on a review of relevant literature, including previous classification approaches. The argumentation schemes are grouped according to more general categories. They distinguish the three main categories *reasoning* (see examples in Table 10 and Table 11), *source-based arguments* (see examples in Table 12), and *applying rules to cases* (see examples in Table 13). The source of the given examples can be found in the footnote[6].

Depending on the specific context, argumentation schemes may be applied in an inappropriate way. For instance, they may use implicit presumptions that cannot be realistically maintained, or they may be based on rhetorical tricks to deceive the audience or the opposition party. Table 14 presents some of the major fallacies, based on a compilation in Walton (2008, pp. 18–22). The source of the given examples can be found in the footnote[7].

1.8 Summary

The fields of argumentation theory, philosophy of language, and pragmatics provide the theoretical foundation and background of many psychological and educational approaches to argumentation. One can observe a considerable shift from reductionist logico-formal treatments of argumentation to modern approaches that encompass pragmatic aspects of language use. In contrast to formal languages, natural argumentative discourse typically involves many unspoken and implicit elements. This generally poses a stiff challenge to the analysis of discourse and renders strictly formal-logical approaches as unpromising endeavors. A possible explanation for the inexplicitness in language use is that discussants mutually assume a shared pool of common knowledge and behave according to maxims of cooperativeness, which demand, among other things, to produce language in an economic way (i.e., only providing information required for current purposes). If each and every used premise were made explicit, language would be hopelessly verbose and extremely arduous, or impossible, to follow. While the traditional formal-logical perspective uses a very narrow notion of acceptability—formal validity—more modern approaches judge

[6] The examples are originally presented or cited in: [1] – Walton (1996, p. 74), [2] – Walton et al. (2008, p. 9), [3] –Walton et al. (2008, p. 10), [4] – Walton et al. (2008, p. 26), [5] – Walton et al. (2008, p. 118).

[7] The examples are originally presented or cited in: [6] – (Walton 2005, p. 97), [7] – Walton (2008, p. 40), [8] – Walton (2008, p. 84), [9] – Walton (2008, p. 21), [10] – Walton (1999, p. 71), [11] Walton (2008, p. 19), [12] – Woods and Walton (1977, p. 584), [13] – Walton (2008, p. 244), [14] – Walton (2008, p. 22–23).

many inferences that are defeasible and presumptive as acceptable as well. For instance, the notion of *informal fallacies* is nowadays widely adopted, which focuses on discussion moves that impair the quality of discourse. Such discussion moves may not only be problematic in terms of their relevance, consistency, and logic. Often, they also employ rhetorical tricks and attempts of willful deception. Argumentation is field-dependent rather than universal. The specific context of argumentation is thus of major importance. This chapter discussed six prototypical types of discourse, proposed by Walton, which are based on different standards and procedures. The decision whether a certain inference is legitimate or an informal fallacy must take the specific context into consideration. For instance, an argument from expert opinion may be legitimate only in some cases. On the one hand, expert opinions are an accepted source of evidence in lawsuits (which may lead to excesses such as a "battle of experts"). On the other hand, when opposition against an expert opinion is generally turned down without further consideration, this may be viewed as the informal fallacy of *ad verecundiam*, or *appeal to modesty*. Argumentation theory seems to converge at viewing a critical discussion, or persuasion dialogue, as a particularly valuable form of discourse. Such dialogues are characterized by an orientation towards standards of reasonableness and fairness in the resolution of a conflict of opinions.

Table 10

Exemplary argumentation schemes in Walton's category *reasoning – deductive and inductive schemes*

Category	Example	General reasoning pattern (schematic)
Deductive reasoning (ex: Reductio ad absurdum)	There is no smallest rational number greater 0. Let's assume such a number would exist, and name this number *r*. Then *r*/2 would be a rational number greater 0 and smaller than *r*. This contradicts the assertion that *r* is the smallest rational number greater 0.	P_1: There is a sequence of deductively valid inference steps from proposition p_1 to proposition p_2 ($p_1 \Rightarrow \ldots \Rightarrow p_2$) P_2: Proposition p_2 is inconsistent with p_1. *C*: Therefore, p_1 must be false.
Inductive reasoning (ex: Argument from random sample to a population)	*Complex argument in scientific paper including a description of hypotheses, operationalization of variables, sampling procedure, experimental design, statistical analysis, results, conclusions, etc.*	P_1: Sample *S* is representative for population *P* (according to the rules of statistical sampling). P_2: Analysis of *S* through proper application of inferential statistics results in assertion *A* about *P* at a confidence level of *C*. P_3: *C* is considered sufficiently large to warrant a statistical generalization. *C:* Therefore, assertion *A* describes a general pattern in population *P*.

Table 11

Exemplary argumentation schemes in Walton's category *reasoning – presumptive schemes*

Category	**Example**	**General reasoning pattern (schematic)**	**Critical questions (selection)**
Causal reasoning (ex: Argument from cause to effect; Walton et al. 2008, p. 328)	"When nations do not remain consistent in their policies, their prestige drops. Recognition of communist China means not remaining consistent in our policies. Therefore, our recognition is likely to drop." [1]	P_1: "Generally, if *A* occurs, then *B* will occur." P_2: "In this case, *A* occurs." C: "Therefore, in this case, *B* will occur."	"How strong is the causal generalization?" "Is the evidence cited (if there is any) strong enough to warrant the causal generalization?" "Are there other causal factors that could interfere with the production of the effect in the given case?"
Abductive reasoning (ex: Argument from sign; Walton et al. 2008, p. 329)	"These look like bear tracks, so a bear must have passed along this trail." [2]	P_1: "*B* is generally indicated as true when its sign, *A*, is true." P_2: "*A* is true in this situation." C: "Therefore, B is true in this situation."	"What is the correlation of the sign with the event signified?" "Are there other events that would more reliably account for the sign?"
Practical reasoning (ex: Argument from waste; Walton et al. 2008, p. 326)	"A PhD student […] has spent more than five years trying to finish her PhD thesis, but there are problems. […] She contemplates going to law school, where you can get a degree in a definite period. But then she thinks: 'Well, I have put so much work into this thing. It would be a pity to give up now.'" [3]	P_1: "If person *P* stops trying to realize goal *G* now, all *P*'s efforts to realize *G* will be wasted." P_2: "If all *P*'s previous attempts to realize *G* are wasted, that would be a bad thing." C: "Therefore, P ought to continue trying to realize *G*."	"Is bringing about *G* possible?" "Forgetting past losses that cannot be recouped, should reassessment of the cost and benefits of trying to bring about *G* from this point in time to be made?"

Table 12

Exemplary argumentation schemes in Walton's category *source-based arguments* (presumptive)

Category	Example	General reasoning pattern (schematic)	Critical questions (selection)
Argument from position to know (ex: Argument from expert opinion; Walton et al. 2008, p. 19, p. 310)	"James W. Johnston, chairman of R. J. Reynolds Tobacco Company, testified before Congress that tobacco is not an addictive substance and that smoking cigarettes does not produce any addiction. Therefore, we should believe him and conclude that smoking does not in fact lead to any addiction." [4]	P_1: "Source *E* is an expert in subject domain S containing proposition *A*" P_2: "*E* asserts that proposition *A* (in domain *S*) is true." P_3: "If source *E* is an expert in subject domain *S* containing proposition *A*, and *E* asserts that *A* is true, then *A* may plausibly be taken to be true." C: "Therefore, *A* may plausibly be taken to be true."	"How credible is *E* as an expert source?" "Is *A* consistent with what other experts assert?" "Is *E*'s assertion based on evidence?"
Argument from popular acceptance (ex: Argument from popular opinion; Walton et al. 2008, p. 311)	Most people believe in god, in one form or the other. Therefore, god exists.	P_1: "If an assertion *A* is generally accepted as true, that gives a reason in favor of *A*." P_2: "*A* is generally accepted as true." C: "Therefore, there is reason in favor of *A*."	"What evidence like a poll or an appeal to common knowledge supports the claim that *A* is generally accepted as true?" "Even if *A* is generally accepted, are there any good reasons for doubting that *A* is true?"

Table 13

Exemplary argumentation schemes in Walton's category *applying rules to cases* (presumptive)

Category	Example	General reasoning pattern (schematic)	Critical questions (selection)
Arguments based on cases (ex: Argument from analogy; Walton et al. 2008, p. 9, p. 200, pp. 58–62)	"When a murderer is found guilty, he is punished regardless of his reasons for killing. Similarly, anyone partaking in an abortion is guilty of having deprived an individual of her or his right to life." [4]	P_1: "Generally, case C_1 is similar to case C_2." P_2: "The similarity between C_1 and C_2 observed so far is relevant to the further similarity that is in question." P_3: "Assertion *A* is true in case C_1." C: "Therefore, *A* is (plausibly) true in case C_2."	"Are there important differences (dissimilarities) between C_1 and C_2?" "Is there some other case C_3 that is also similar to C_1, except that *A* is false in C_3?"

Verbal classification arguments (ex: Argument from verbal classification; Walton et al. 2008, p. 319)	"Marcia and Ted are arguing on the issue of abortion. Ted, who is prolife, argues: 'There can be no abortion when the fetus becomes a person.' Marcia replies: 'That's hopelessly vague! There is no way to exactly define when the fetus has become a *person*. You don't leg to stand on there!'" [5]	P_1: "*A* has property *f*." P_2: "For all *X*, if *X* has property *f* then *X* can be classified as having property *g*." C: "Therefore, *A* has property *g*."	"What evidence is there that *A* definitely has property *f*, as opposed to evidence indicating room for doubt about whether it should be so classified?" "Is the verbal classification in the classification premise based merely on an assumption about word usage that is subject to doubt?"

Table 14

Major argumentation fallacies

Fallacy	Example	Description / Comment
Fallacy of arguing in cycles	"Ella: God exists. Brad: How do you know? Ella: The Bible says so. Brad: How do I know what the Bible says is true? Ella: Because the Bible is the word of God." [6]	Justifying a conclusion with a premise that follows from the conclusion. The argument therefore has no force in supporting the conclusion. From a formal-logical point of view, such arguments are valid.
Fallacy of complex question	"Have you stopped abusing your spouse?" [7]	Using a question that contains a presupposition to fool the addressee. No matter how the addressee answers, he will implicitly accept the presupposition. Thus, the addressee may reject the question.
Wrong conclusion / red herring fallacy	"The prosecuting attorney in a criminal trial is supposed to prove that the defendant is guilty of murder. However, the prosecuting attorney argues at length that murder is a horrible crime. He holds up the victim's bloody shirt for the jury to see. He expostulates at length on the horror of this crime and all crimes of murder." [8]	Fallacies of irrelevance. The example can be interpreted as "wrong conclusion" fallacy (argument supporting a conclusion different to the one in question) or "red herring" fallacy (the argument wanders off, does not arrive at any particular conclusion at all, distracts the audience). If the presented argument turned out to be relevant to the conclusion in the course of the trial, the argument would not be classified as fallacy.

Fallacy of equivocation turns	“All elephants are animals, and Henri is an elephant, and Henri is a small elephant; therefore Henri is a small animal.” [9]	Same term is used with different, confusing meanings. In the example, the first occurrence of “small” is meant relative to elephants, the second occurrence of “small” is meant relative to animals. The confusion is that a small elephant is still a relatively big animal.
Appeal to emotions	Real-estate salesman to potential customer: “A man of your standing in the community cannot afford to live in a cheap neighborhood.” [10]	Disguising a lack of strong and relevant evidence by appealing to the emotions of the audience (popular sentiments, group solidarity, pity, fear). The example demonstrates an argument that uses flattery and appeal to prejudice and snobbery to persuade a potential customer.
Argumentum ad hominem	“The so-called theories of Einstein are merely the ravings of a mind polluted with liberal, democratic nonsense which is utterly unacceptable to German men of science.” (Dr. Walter Gross, official representative of “Nordic Science” in Nazi Germany) [11]	Personal attacks, off the point, aimed at discrediting an opponent. Arguments referring to a person are not per se a fallacy, in particular, if personal conduct and character are in question.
Post hoc, ergo propter hoc	“I took a dose of Sinus Blast and a couple of days later my cold cleared right up.” [12]	Wrongly concluding that an event *B* is caused by an event *A* because event *B* occurred after event *A* (chronologically).
Appeal to modesty	“Well, look, you’re not an expert. So nothing you can say about the matter is anything less than presumptuous.” [13]	Trying to unfairly and dogmatically “silence the opposition” against an assertion by pointing to the authority of an expert. Opposition is viewed as a lack of respect for the opinion of the expert.
Straw man fallacy	“C: It would be a good idea to ban advertising beer and wine on radio and television. These ads encourage teenagers to drink, often with disastrous consequences.” “A: You cannot get people to give up drinking; they’ve been doing it for thousands of years.” [14]	Misrepresenting an argument of an opponent (misquote, exaggerate, etc.) and attack the opponent based on the misrepresented argument. The example illustrates this kind of fallacy, assuming that person C does not argue in favor of getting people generally to give up drinking during the dialogue.

Chapter 2
Argumentation Research in Psychology and Education

The research presented in the previous chapter is mostly based on theoretical considerations and the detailed analysis of individual examples of argumentative discourse. Sometimes, these examples are real but often they are fictional, devised by scholars to illustrate the main aspects of their theories or show how their systemizations can even account for particularly pathologic or borderline cases. This chapter discusses research in psychology and education with a main focus on empirical results, most of which are quantitative in nature.

Section 2.1 addresses two important questions: What skills are foundational for argumentation, and to what extent do people possess these skills? To answer these questions, results from cognitive psychology regarding people's formal reasoning skills are discussed (e.g., inferences of the modus ponens type). While formal reasoning certainly plays a role in constructing valid arguments, it takes much more to be competent in argumentation. Quite the contrary, argumentation is typically associated with *informal reasoning*. Rather than performing the one expected correct inference step, argumentative reasoning becomes relevant when situations are complex and unclear, when multiple alternatives exist, and when information is uncertain and incomplete. Corresponding research on informal reasoning and the skills of argument will be reviewed. Section 2.2 discusses cognitive models of argumentation. Two important questions are addressed: Are argumentation skills domain-general or domain-specific? And: What role does domain-specific knowledge play with respect to argumentative performance? Section 2.3 discusses epistemic beliefs, which are the theories people hold regarding the nature of knowledge and knowing. As will be discussed, such theories are assumed to have a major influence on people's attitude towards argumentation and thus, their argumentative performance. Section 2.4 reviews results from developmental psychology regarding the development of argumentation skills from childhood to adult age. An important finding is that education is a crucial factor in the development of argumentation skills. Yet, as discussed in section 2.5, current educational practice is not particularly successful in fostering argumentation skills. Therefore, educational research has identified argumentation as a field of study of high practical relevance. In particular, social learning approaches have attracted much interest. Section 2.6 gives a general overview of social learning theories and research including a discussion of expert tutoring and cooperative learning. Section 2.7 discusses argumentation-based learning approaches and corresponding empirical results grouped according to three main categories of targeted or achieved learning outcomes: learning to argue, arguing to learn, and arguing to improve thinking. Section 2.8 summarizes the main insights of this chapter.

2.1 The Skills of Argument

This section discusses research on cognitive skills relevant to the production, interpretation, and evaluation of arguments. While some relevant results regarding formal reasoning skills are discussed, the bulk of this section is devoted to research on informal reasoning, which is focused on the forms of argumentation employed to address complex real-world problems. Specific argumentation skills and empirical results that indicate widespread deficiencies in using these skills are presented.

Results from cognitive psychology indicate that many people have limited abilities in formal deductive reasoning and probabilistic judgment. Table 15 shows formal-logical inferences and corresponding fallacies investigated in cognitive psychology studies. Marcus and Rips (1979) found that almost all subjects were able to correctly make *modus ponens* inferences. However, only about half of them correctly draw *modus tollens* inferences. Moreover, many subjects committed formal fallacies of the types *affirming the consequent* and *denying the antecedent*. Byrne (1989) found that the context can have a strong effect on the making of correct and incorrect inferences. For instance, providing multiple if-then statements with the same conclusion helps subjects realize that the if-part only gives a sufficient but not a necessary condition. Other studies show problems in making probabilistic judgments. For instance, Tversky and Kahneman (1974) used a task in which subjects estimated the probability that a person has a certain occupation based on a description of that person. In addition, information regarding the base-rate frequencies of different occupations was provided. Even if the description was completely uninformative, subjects ignored the provided base-rate frequencies in their estimates (*base-rate fallacy*). If no additional descriptions were provided, subjects correctly used the given prior probabilities in their estimates. A possible explanation for formal and probabilistic reasoning errors is that in many everyday situations, simpler but fallible reasoning strategies are sufficient for current purposes and more economic, since they require less thinking time and mental efforts (*bounded rationality*; Gigerenzer et al. 1999; Kahneman 2011; Simon 1955).

Table 15

Formal inferences and fallacies used in cognitive psychology studies on formal reasoning

Inference type	**Reasoning pattern**	**Example**
Modus ponens (formally valid inference)	If P then Q. P. *Therefore*, Q.	"If she has an essay to write then she will study late in the library." "She has an essay to write." "*Therefore*, she will study late in the library."
Modus tollens (formally valid inference)	If P then Q. Not Q. *Therefore*, not P.	"If she has an essay to write then she will study late in the library." "She will not study late in the library." "*Therefore*, she does not have an essay to write."
Affirming the consequent (formal fallacy)	If P then Q. Q. *Therefore*, P.	"If she has an essay to write then she will study late in the library." "She will study late in the library." "*Therefore*, she has an essay to write."
Denying the antecedent (formal fallacy)	If P then Q. Not P. *Therefore*, not Q.	"If she has an essay to write then she will study late in the library." "She does not have an essay to write." "*Therefore*, she will not study late in the library."

Note: Examples taken from Byrne (1989).

While formal reasoning problems can be characterized as "well-structured, familiar, and compatible with existing knowledge," informal reasoning problems are generally "open-ended, debatable, complex, or ill-structured" (Means and Voss 1996). They typically involve the generation or evaluation of evidence and reasons vis-à-vis one or more claims in question. Argumentation is thus central to informal reasoning. A pioneering empirical study on informal reasoning is presented by Kuhn (1991). The study focused particularly on argumentation skills, which Kuhn considers pivotal to human thinking as such. Thinking often takes the form as arguing with oneself, e.g., to form beliefs, to make judgments, and to make decisions. The study investigated whether common people (i.e., non-experts) are able to rationally justify their beliefs about controversial topics of general interest. While the study investigated individual reasoning, the results also provide some insights with respect to collaborative argumentation. Following Kuhn, collaborative argumentation employs essentially the same set of elementary argumentation skills (plus some additional ones, e.g., socially-oriented skills).

The study consisted of research interviews with individual participants about three social problems (e.g., "What causes prisoners to return to crime after they're released?"). In the course of the interview, participants were asked a number of questions to assess specific argumentation skills. In particular, participants were asked to do the following things (in parentheses the specific skill addressed):

1. present their own causal theories on these issues (*theory*)
2. justify their causal theories and give supporting evidence, if possible (*evidence*)
3. generate possible counter-positions (*alternative theories*)
4. devise counterarguments against their own causal theory (*counterarguments*)
5. rebut possible counterarguments (*rebuttals*)
6. evaluate evidence presented by the interviewer (*evaluation of evidence*)

With respect to (1), the study showed that people have coherent causal theories about questions of general interest, although the complexity of the presented theories varied (single-cause and multi-cause theories at different levels of elaboration). With respect to (2), three broad categories were distinguished: *genuine evidence*, *pseudo-evidence*, and *no evidence*. In contrast to genuine evidence, pseudo-evidence only illustrates or elaborates on the causal mechanism, that is, how the cause brings about the effect. Pseudo-evidence may make a theory clearer, more interesting, or more plausible, but does not give any indication of its correctness. Therefore pseudo-evidence may be considered as no evidence at all but rather as a part of the theory itself. The study showed that exactly such pseudo-evidence was the prevalent form of evidence used. With respect to (3), the majority of participants were able to generate alternative theories. Interestingly, those who were able to do so were also more likely to generate genuine evidence. Being aware of the existence of alternative theories is an important prerequisite to understanding that one’s own theory might be wrong. In the light of multiple opposing theories, the importance of genuine evidence, as a criterion to determine which theory is preferable, becomes most apparent. With respect to (4), many participants were not (fully) successful in generating counterarguments. While some offered at least an alternative theory when prompted for a counterargument (considered by Kuhn as a partial success), others failed or declined to produce a counterargument. Anticipating possible counterarguments is important to identify weak spots of one's own theory. Apparently, many participants were not able or willing to critically reflect on their own position, a skill central to argumentation. With respect to (5), three main sorts of rebuttals were distinguished: (a) rebuttals that try to undermine the force of the counterargument (showing that the counterargument is of no or only limited use to attack the original argument), (b) rebuttals that try to establish that one's own theory (or a slightly adjusted version of it) is nonetheless more correct than alternative theories, and (c) rebuttals that simply argue against the alternative theory without referring to one's own theory. The study showed that many participants did not successfully generate rebuttals. When rebuttals were successfully generated, they were often of the last mentioned type: simple rebuttals. Such rebuttals are solely focused on the alternative theory, and thus, can be generated without critically reflecting possible deficiencies of one's own theory. With respect to (6), participants were presented with under-determined evidence (in particular: a description of a single instance of a phenomenon with only minimal cues regarding possible causes) and over-determined evidence (in particular: a description of three studies each clearly indicating a different causal explanation). The responses showed that many participants did not make a clear distinction between their own theory and the presented external evidence. Rather, the presented evidence was perceived and interpreted (sometimes misinterpreted) through the lens of their own theory. In the case of under-determined evidence, participants often explained the presented phenomenon in terms of their own theories. In the case of over-determined evidence, participants typically focused on the parts that corresponded with their theory while ignoring other parts that may disprove their theory.

In summary, the results indicate clear deficiencies of many participants with respect to important argumentation skills, such as generating genuine evidence, anticipating and rebutting possible counterarguments, and critically and objectively interpreting external evidence. As noted by Kuhn (1991, p. 282), these results are consistent with other investigations of informal reasoning skills that identified a *myside bias* (i.e., focusing on evidence and reasons that support one's own beliefs while ignoring disconfirming information; Perkins et al. 1983) and a *make sense epistemology* (i.e., analyzing situations only to the point that a mental model is found that superficially makes sense without further critical reflection on it; Perkins 1985) in people's reasoning.

2.2 Cognitive Foundations of Argumentation

An important question is whether the performance in argumentative tasks is based on a general skill (*domain-generality hypothesis*) or on knowledge in specific domains (*domain-specificity hypothesis*). The results of Kuhn (1991) suggest that both factors may play a role. On the one hand, the quality of argumentation varied depending on the specific topic. On the other hand, there was also some consistency in the participants' performances across topics. So while domain-specific factors may play some role, there also seems to be a general, domain-independent reasoning component at work.

In line with this observation and their own data, Means and Voss (1996) propose a two-component model of informal reasoning comprising a general *informal reasoning component* and a *knowledge-experiential component*. The model can be described in relation to the *construction-integration model* of text comprehension developed by Kintsch and Van Dijk (see, for instance, Kintsch 1994). The construction-integration model explains the comprehension of texts in terms of two interwoven processes: the construction of a propositional cognitive representation of given textual inputs and the integration of this representation into a propositional network of prior knowledge. Conceptually, two kinds of cognitive structures may be distinguished: The *textbase* represents all the elements and relations contained in the given textual input. The *situation model* integrates the textbase with prior knowledge and enhances it with further elaborations inferred from text and prior knowledge. The textbase can be associated with a shallow level of understanding (or remembering), which is typically sufficient for reproducing the content of the text. The situation model can be associated with deep understanding (or learning). The level of elaboration of the situation model decides how productively the content of the text can be leveraged in novel environments. The informal reasoning component proposed by Means and Voss (1996) embodies conventions of reasoning, which are reflected in language patterns used by relatively educated people. If the component is highly developed, people are able to construct elaborate situation models from given textual inputs, that is, situation models that include and enable advanced inferences about the given textual input (e.g., comparing different reasons for quality or adequateness; anticipating counterarguments). If the component is not sufficiently developed, people are only able to construct impoverished situation models. Their inferences therefore go not much beyond what is represented in the textbase itself. The effect is cumulative, since new situation models are constructed in the context of existing ones (i.e., a richer or poorer knowledge base can be exploited when processing new textual input). This is where the knowledge-experiential component enters the picture: Constructing a high-quality situation model requires, in addition to informal reasoning skills, a sufficient amount and elaboration of relevant prior knowledge. The importance of prior knowledge is also highlighted in other research (e.g., Sadler and Zeidler 2005; Von Aufschnaiter et al. 2008).

Domain-general thinking skills may not be immediately applicable to domain-specific knowledge. Rather, as proposed by Perkins and Salomon (1989), domain-general skills may have to be contextualized to different knowledge domains for effective use. In reference to Toulmin's (1958) notion of field dependence, they point out that the general structures of arguments in different domains resemble one another considerably. However, criteria for eligible evidence typically differ between domains. For instance, a lawyer may not be good in producing or evaluating scientific arguments, and likewise a scientist may have trouble with legal arguments. So higher-order knowledge and skills of good argumentation in general may be complemented with (or specialized into) more specific knowledge and skills that consider the rules and patterns of argumentation in particular domains. On the most-specific level, then, content knowledge elements of the domain in question may enter the equation as the objects on which the more general skills operate. If argumentation is conceived of as a means to constructing knowledge, domain-specific argumentation skills can be the basis of engaging in what Morrison and Collins (1996) refer to as *epistemic games*, that is, culturally patterned ways of constructing knowledge. They identify *epistemic fluency* as an important objective of education. Students should learn to recognize and engage in a large number of relevant epistemic games, each of which typically including some domain-dependent form of argumentation.

Anderson, Reznitskaya and colleagues (Anderson et al. 2001; Reznitskaya et al. 2009) propose a schema-based account of argumentation skills, which also distinguishes different level of specificity of argumentative knowledge. They use the term *argumentation schema* to refer to a knowledge system responsible for organizing and retrieving argument-related information, deciding when to engage in argumentation, and constructing, anticipating, and evaluating arguments. It includes declarative knowledge (*knowing about* argument components, etc.) and procedural knowledge (*applying* argumentative skills). The argument schema represents knowledge in an abstract and generalized way so that argumentation skills can be applied across different situations. The argument schema includes *argument stratagems*, which represent recurring patterns of argumentation. A stratagem comprises information about its purpose and function, associated language forms, conditions of use, possible kinds of objections, etc. Anderson et al. (2001) use a template-based notation to label stratagems. For instance, "What if [SCENARIO]?" describes a stratagem that can be used to test some conclusion against a hypothetical scenario. The term in brackets is a template that varies depending on the specific situation of use. While the deep semantic meaning of the stratagem is stable, different language forms may be used to express the stratagem. Anderson et al. (2001) interpret the use of varying language forms as an indication that students grasp the deep meaning of the stratagem rather than just copying the surface structure. Generally, deep understanding of a stratagem is indicated by appropriate and repeated use of different associated language forms across different contexts. Anderson et al. (2001) propose that humans develop an argument schema through sustained participation in *collaborative reasoning*. Corresponding instructional approaches will be discussed below.

Overall, four conclusions may be drawn from research on the cognitive foundations of argumentation:

1. Deficiencies in informal reasoning and argumentation skills may have detrimental effects on text comprehension and content learning [argumentation skills → content learning].
2. High quality informal reasoning and argumentation performance requires a sufficient level of background knowledge in the domain under discussion [argumentation skills + content knowledge → argumentative performance].
3. While argumentation skills may be to some extent domain-independent, they nevertheless have to be contextualized to be (fully) functional in specific domains.
4. Language seems to play an important role in the development of argumentation skills.

2.3 Epistemological Beliefs and Thinking Dispositions

The argumentation skills investigated by Kuhn (1991) may be seen as joint contributors to a more general ability, "the ability to *contemplate* whether what one believes is true, in contrast simply to *knowing* that it is true" (p. 264, emphasis in original). Hence, a relevant factor in developing and using argumentation skills may be people's general stance towards the process of knowing. Kuhn investigated the relationship between argumentation skills and *epistemological theories* that people hold, that is, their views on the nature of knowledge, beliefs, and evidence. Such theories constitute meta-level knowledge that drives belief formation, evaluation, and revision. Table 16 shows the four different levels of epistemological understanding.

Table 16
Levels of epistemic understanding

Level	Assertions	Reality	Knowledge	Critical thinking
Realist	Assertions are *copies* of an external reality.	Reality is directly knowable.	Knowledge comes from an external source and is certain.	Critical thinking is unnecessary.
Absolutist	Assertions are *facts* that are correct or incorrect in their representation of reality (possibility of false belief).	Reality is directly knowable.	Knowledge comes from an external source and is certain.	Critical thinking is a vehicle for comparing assertions to reality and determining their truth or falsehood.
Multiplist	Assertions are *opinions* freely chosen by and accountable only to their owners.	Reality is not directly knowable.	Knowledge is generated by human minds and is uncertain.	Critical thinking is irrelevant.
Evaluativist	Assertions are *judgments* that can be evaluated and compared according to criteria of argument and evidence.	Reality is not directly knowable.	Knowledge is generated by human minds and is uncertain.	Critical thinking is valued as a vehicle that promotes sound assertions and enhances understanding.

Note: Adapted from Kuhn et al. (2000).

As described in Kuhn et al. (2000), the epistemological theories people hold may be seen as the result of a developmental process, in which a balance is sought between the two dimensions of knowing: objectivity and subjectivity. Typically until the age four, children hold the *realist* view. At this level, knowledge is seen as an exact copy of the external world and therefore certain and unarguable. When children begin to realize that people have different, sometimes wrong beliefs about the world, they are about to enter the *absolutist* level. They still assume that an absolute, objective truth exists. So, the reason for different beliefs must be that some people are lacking information or simply misunderstand the world. During the transition from the absolutist to the *multiplist* level, people increasingly gain the insight that knowledge is the product of the human intellect. Every individual has her own history and made her own experiences, so it is not surprising that people interpret the world differently and arrive at different legitimate views and opinions. While this perspective marks an important advancement from the absolutist level, it also entails an indifferent and uncritical attitude towards opinions. Every opinion is essentially equally right, no matter how substantial the reasons speaking against it are. Finally, some people may reach the highest level of epistemological understanding, the *evaluativist* view, in which standards of objectivity reenter the picture. There are criteria to evaluate and compare opposing theories in an objective and reasonable way. While absolute certainty is an unattainable goal, it is nonetheless possible to judge whether theories are logically consistent, whether some theories are more likely than others in the light of given evidence, or whether the empirical basis is simply too weak to give a reliable judgment.

Kuhn (1991) classified the participants according to the last three levels (the first level is irrelevant since typically only small children hold that view). People who expressed that experts have, or principally can gain, incontestable knowledge about a question were classified as absolutists. Persons who denied this possibility and furthermore claimed that they personally know better or equally well as experts about the topic in question were classified as multiplists. Finally, persons who denied absolute expert certainty while conceding that experts can judge the given question with greater certainty as they can have been classified as evaluativists. The study results indicated that evaluativists, who comprised only a relatively small portion of all participants, indeed exhibited superior argumentation skills in comparison to both absolutists and multiplists. Similar results have been reported by Mason and Scirica (2006).

Epistemological beliefs may be one factor contributing to a person's *thinking disposition*. The concept of thinking dispositions (Perkins et al. 1993; Perkins et al. 2000) is motivated by the observation that a poor performance, particularly in informal reasoning tasks, is not necessarily caused by a lack of ability. Rather, participants may be principally able to successfully engage in some desired form of thinking but do not do so for other reasons. Perkins et al. (2000) present evidence in favor of a three-component model of thinking dispositions. First, a person must be able to carry out a certain behavior (*ability*). Second, a person must recognize opportunities to make use of this ability (*sensitivity*). Third, a person must decide to take the opportunity to make use of the ability (*inclination*). Motivational factors play a role with respect to inclination, but also considerations of whether investing one's intelligence into some issue is worth the effort. Perkins et al. (2000) found in a study that in more than 86% of all cases, a lack of sensitivity prevented people from engaging in forms of thinking they are principally able to carry out. Inclination accounted for about 55% of situations in which subjects did not make use of thinking abilities they have. People who do not hold an evaluativist view can be expected to see less value in critical thinking. In consequence, they may be less sensitive to critical thinking opportunities and less inclined to engage in critical thinking activities. This may not only affect the observable performance but also hamper learning and development since opportunities to practice critical thinking skills are missed. Vice versa, holding an evaluativist view may directly contribute to the development of critical thinking and argumentation skills, which is in line with the results of Kuhn (1991).

2.4 Development of Argumentation Skills

As discussed, the epistemological stance of a person develops over time and only sometimes reaches the evaluativist level, which is conducive to productive argumentation. But what is known about the development of argumentation skills themselves?

Berkowitz and Gibbs (1985) compared the extent of *transactive moves* used in different age groups (ages between 6 and 20 years) during moral conflict discussions. Transactive discussion moves are defined as moves that involve the reasoning on the reasoning of others. They may be further subdivided into *representational transacts*, which, by and large, only re-present the reasoning of others (e.g., paraphrasing), and *operational transacts*, which significantly transform the reasoning of others (e.g., counter-arguing). Particularly operational transacts may be considered as indicators for quality discussions and associated with a high skill level in argumentation. The study showed that the amount of transactive moves grows as a function of age. In the group with the youngest participants (6–8 years), transactive moves were almost completely absent. The biggest leap was observed from the age group 12–14 to the age group 15–17, which suggests that the period of early to middle adolescence is critical in the development of argumentation skills. Another study reported by Berkowitz and Gibbs (1985) found that the formal operational stage of Piaget's theory of cognitive development (1972) is strongly associated with transactive discussion. The formal operational stage is characterized by logical and abstract forms of thinking, including deductive and hypothetical reasoning, which may be seen as prerequisites for proper argumentation. The transition to the formal operational stage is supposed to happen between adolescence and adulthood, that is, exactly at the time when the biggest increase in transactive talk was observed in the first Berkowitz and Gibbs (1985) study reported here.

Kuhn (1991) locates the main period of argumentation skill development slightly earlier on the timeline. The study originally reported in Kuhn (1991) did not yield any significant differences between the youngest age group (9th grade, i.e., ages ranging between 14 and 15) and older age groups. However, a previous study of Kuhn and colleagues (1988), which investigated similar abilities (production of evidence and counterarguments, evaluation of evidence) in a similar, yet somewhat simpler setting, showed a major improvement occurring just before that age, across the preadolescence age and the early adolescence age. From then on, the educational level seems to be the main factor. Only the group of young adults who attended college for several years displayed further performance improvements.

Stein and Miller (1993) judge the argumentation skills of children even more favorably than Kuhn (1991) and Berkowitz and Gibbs (1985). They found that children at the age of seven are already able to display argumentation skills that are in many respects similar to the ones of college students (e.g., in terms of relevance, coherence, and logic). Children in this age group were already able to identify and use basic argument components to justify and evaluate positions. Stein and Miller (1993) identified the value system children and adults hold as an important differentiating factor. For instance, the dilemma problem used in their study forced a decision between two opposing options: “sticking to an agreement” versus “not threatening the friendship.” While most adults valued the former option more, the majority of children valued the latter option more. This was not only visible in the initially chosen position but also in how participants evaluated potential arguments of both sides. Stein and Miller (1993) attribute previous results, which purport that children’s arguments are personalized and illogical, to problematic materials and research methods. In particular, they criticize that the used materials have often been developed for adults rather than for children. Children may have not been able to display competent argumentation skills because they were simply lacking appropriate background knowledge and understanding of the domain under discussion. Furthermore, the quality of argumentation invoked may strongly depend on situational demands. If no critical and probing questions are being asked, a person may assume that her claims have been accepted and refrain from presenting further or better support. Finally, it is important to consider the specific goals and agenda a person pursues in an argument. If a person approaches a discussion in a conflicting rather than a cooperative way (i.e., trying to win at any costs), the quality of the used arguments may suffer. In summary, there is a difference between the arguments a person actually produces and the arguments a person is principally able to produce. Stein and Miller (1993) see the circumstantial conditions of many previous studies as insufficient to allow children to display the real extent of their argumentation skills.

Felton and Kuhn (2001) compared the discourse strategies of adults and adolescents. As a theoretical framework they used Walton's model of a critical discussion (2008; see also section 1.7). In this type of dialogue, discussants try to elicit commitments from fellow discussants and build their own argumentation upon these commitments. That is, if a discussion partner has committed to some statements and one can derive one's own conclusions from these accepted statements, then the discussion partner is also committed to accept these conclusions, if obeying the norms of reasonableness. From this observation, two discourse goals can be derived: (1) getting the partner to concede to premises that are central to one's own argumentation, and (2) identifying and challenging questionable premises (implicit or explicit) in the partner's argumentation. Different discourse strategies may be utilized to realize these goals. The development of argumentive discourse skills therefore involves two interrelated processes: (1) developing an understanding of the goals involved in argumentative discourse, and (2) developing strategies to effectively achieve these discourse goals. In a study, Felton and Kuhn (2001) found that adults use more powerful argumentative strategies than adolescents. For instance, adults used more counterargument moves. They also pursued effective multi-turn strategies more frequently, such as eliciting, in a targeted way, commitments from the discussion partner that can be used to attack and undercut the partner's position. Some evidence also indicated that adults have a better understanding of discourse goals. Felton and Kuhn (2001) compared the participants' behavior between agreeing and disagreeing dialogues. The initial situation of a disagreeing dialogues are opposing positions, while in an agreeing dialogue, the discussing partners share the same position. While adolescents displayed similar behavioral patterns in both types of dialogues, adults apparently adopted discourse goals particularly suitable for the current type of dialogue. In particular, in agreeing dialogues, adults did not use a strategy of weakening and undercutting the partner's argumentation, which they exhibited in disagreeing dialogues.

In summary, the empirical studies come to different results with respect to the main period of argumentation skill development. Stein and Miller (1993) attest that children at the age of seven already have argumentation skills that are in many respects comparable to the ones of college students. Kuhn et al. (1988) identifies the period from preadolescence to the early adolescence as critical in the development of argumentation skills. Berkowitz and Gibbs (1985) conclude that major improvements occur in early to middle adolescence. The results of Felton and Kuhn (2001) indicate a developmental leap between adolescence and adulthood. To reconcile these results, it is instructive to look at the specific settings and methods used. The studies essentially evaluated different facets of argumentation. The results presented by Stein and Miller (1993) show that young children are already able to produce relevant, coherent, and logical arguments. Yet, the used dilemma problem required argumentation of a rather simple kind, restricted to justificatory reasons for a preferred action ("I'm in favor of action X because …"). Kuhn et al. (1988) used much more demanding problems involving arguments relating to causal claims, which are structurally more complex ("I believe that X causes Y because …"). The standards to judge the quality of arguments were based on a scientifically oriented model of argumentation and thus much more stringent. Such a model requires a proper understanding of the difference between theory and data and the coordination between elements at these two levels. Berkowitz and Gibbs (1985) evaluated argumentation skills in a discussion setting, which again is more demanding in other respects, since discussants must manage multiple tasks at once. In particular, they must advance their own argumentation, parry off critiques of others, follow others' argumentation, and offer well-reasoned critiques. Finally, Felton and Kuhn (2001) focused on one particularly advanced aspect of argumentative discussions, namely the use of strategic behavior, which involves selecting and implementing own discourse strategies as well as recognizing strategic moves of others.

Generally, as noted in Kuhn (1991), it unrealistic to expect that a single point in time can be identified at which a previous absent skill comes suddenly into existence. Rather, argumentation skills develop gradually, on a continuum from the most rudimentary forms to mature and fully functional argumentation skills. At the low end, argumentation skills exist mostly in implicit form and only become visible in environments that provide a high degree of guidance and support. As development proceeds, the amount of external support required to make competent use of argumentation skills continually declines. The learner becomes increasingly autonomous in her performance. At the high end, the learner can act fully autonomously, without external support, and display the skills of argument in an explicit and consistent manner. Appropriate educational approaches may be employed to speed up the developmental process and to pave the way to the higher levels of the developmental continuum. Natural maturation processes, e.g., triggered through informal encounters of argumentative practices in everyday situations, may constitute an insufficient condition for reaching these higher levels and additional educational support may be required. This is supported by findings suggesting that college education makes a difference in terms of argumentation skills of adults (Kuhn et al. 1988) (albeit even highly educated people still often struggle with basic critical thinking and sound reasoning skills, cf. Kahneman 2011).

2.5 Rethinking the Role of Argumentation in Education

As discussed, a number of studies of formal and informal reasoning point out severe deficiencies in argumentation-related skills. This raises the question regarding the quality of the teaching of argumentation skill in schools and universities. An obvious conclusion is that current education systems largely fail in promoting argumentation skill development of young people (Kuhn 1991). Education is too narrowly focused on imparting domain-specific knowledge and skills with the result that general abilities in thinking and informal reasoning improve through the years of formal education to a lesser extent as one would expect (Perkins 1985). For instance, scientific knowledge is often presented as an accumulation of undisputable facts without paying due attention to the social practices of science, which heavily rely on argumentation (Driver et al. 2000). Besides neglecting the learning of argumentation skills as such, this educational approach runs the danger of nourishing an absolutist attitude towards knowledge. Another factor is that the liberal and democratic traditions Western societies are based upon—such as the freedom of opinion and the appreciation of diversity and individualism—may mislead to the conclusion that all opinions are equally right and thus do not require a critical examination. Kuhn (1991) sees this "radical, unreasoned relativism" (or extreme relativistic attitude) as one of the causes of a current educational crisis. For instance, for some time now, there are attempts in the United States to integrate elements of the theory of intelligent design in biology classes—a view that rejects major scientific results regarding evolution and instead proposes that life, in all its complexity, must be the result of an intelligent creator or designer. Although the scientific community sees intelligent design as an unscientific, religious viewpoint, prominent conservative politicians support the idea to teach intelligent design as part of the science curriculum. For instance, despite the lack of appropriate scientific support for intelligent design, former U. S. President George W. Bush stated in 2005 that "both sides [evolutionary theory and intelligent design] ought to be properly taught" (Bumiller 2005).

In face of the discussed problems, educational research has called for educational reforms, and in particular, a stronger emphasis on a critical attitude and the skills of argumentation, which are the cognitive and social tools to put this critical attitude into action (e.g., Driver et al. 2000, Ritchhart and Perkins 2005, Kuhn 2005). An important aspect is certainly a cultural change in the classroom, which makes critical thinking, argumentation, and discourse integral parts of the curriculum, highly valued and regularly practiced. Promoting such a classroom culture is the prerequisite for an *enculturation* of students into the practices of critical thinking and argumentation and the formation of a positive attitude towards these activities (cf. Brown et al. 1989; Driver et al. 2000; Resnick et al. 2010; Tishman et al. 1993). To implement such changes in the classroom, specific instructional methods effective in teaching critical thinking, argumentation, and discourse are required.

A prevalent kind of formal education is still didactical teaching. This approach is certainly appealing from an organizational and economic perspective, since it has a relatively low demand on teacher manpower and time in comparison to other approaches discussed below. Yet, from a pedagogical perspective, the approach has important limitations. At the core of didactical teaching are teacher-centered activities, most notably, the presentation of learning content by the teacher. With respect to argumentation skills, didactic lessons may focus on "teaching people *about* good thinking" (Kuhn 1991, emphasis mine). This may include, for instance, aspects such as the general structure and nomenclature of arguments, or characteristics of good and bad arguments (Voss and Means 1991). Didactic teaching has been criticized for the passive role that learners take as mere recipients of information, which often produces knowledge that is isolated, shallow, cursory, and of little practical use. For instance, Brown (1992) refers to *passive learning* and *inert knowledge* as "diseases of schooling." The approach stands in stark contrast to current learning theories (Piaget 1985; Vygotsky 1978), which describe learning as a constructive process that involves the interlinking of new and existing knowledge as well as significant reorganization and transformation of knowledge structures. Such processes are more likely to be triggered when learners actively engage in appropriate tasks. In line with this perspective, Kuhn (1991) suggests to employ practically-oriented learning activities. Particularly approaches inspired by theories of learning in social contexts gained considerable attention over the last couple of decades.

2.6 Learning in Social Contexts

Social learning approaches emphasize the importance of social interactions for learning. The active participation in social activities is well compatible with the constructivist perspective. Moreover, the social context which the activity is embedded in provides a valuable source of feedback and guidance. Other possible advantages include positive motivational effects social contexts potentially have (Chinn and Clark 2013). Learning through social interactions may even be considered as the most natural (and thus particularly effective) form of learning, since the human brain has an inherent social orientation and is highly sensitive to social influences (Mercer 2013). An observation made by scholars from the antiquity to modern times, including Piaget and Vygotsky, is the resemblance of social arguments and individual deliberations (Anderson et al. 2001). In light of this observation, the hypothesis that social contexts and interactions have a major influence on the development of argumentative reasoning skills appears particularly plausible. Two main social arrangements that have been intensively studied in the past are the learning from a more experienced person and learning in groups of peers.

2.6.1 Scaffolding: Learning from a more experienced person

In the first arrangement, the social context is given by a more experienced person, who might be an adult or a more knowledgeable peer. The close interaction between the teacher and the learner is typically characterized as a *scaffolding* process. As described by Pea (2004), the notion of scaffolding was introduced in the 1970s by the research team of Jerome Bruner. Originally, it described mother-child interactions in informal settings in which the mother supports the child in engaging in activities that would otherwise be beyond reach. As the child becomes more competent through continued practice, the mother reduces the level of support until the child is able to perform the activity independently (*fading the scaffold*). The scaffolding process is essentially geared to the developmental progress of the child. Since this kind of scaffolding occurs naturally, without the mother explicitly and formally trained in giving and adapting support, it may be called *natural scaffolding*. Nowadays, the understanding of the term scaffolding is widely extended to also include instructional approaches that enhance learning processes with structure and guidance according to the needs of the learner. Such more formalized (or designed) educational approaches to scaffolding may be called *instructional scaffolding*. As pointed out by Tabak (2004), there is some debate which instructional approaches may actually be considered as scaffolding. A defining criterion one may employ is that scaffolding always provides support for both, practice and learning. Pea (2004) describes different mechanisms of scaffoldings:

1. The degrees of freedom of a task may be reduced and the set of possible actions constrained to avoid undesired and foster desired activities (*channeling*).
2. The learner's attention may be directed towards particularly relevant task features, again, to promote activities important for successful task completion (*focusing*).
3. More advanced solutions to the task may be displayed to the learner (*modeling*).

The notion of scaffolding is related to the Vygotskian concept of the *zone of proximal development*, which comprises all activities a child can only accomplish with the assistance of an adult or a more capable peer. According to the socio-cultural tradition that evolved from Vygotsky's seminal work, *guided participation* in cultural activities gradually leads into autonomous performance of these activities (Rogoff 1990). Activities initially mediated through interpersonal interaction on the social plane are internalized, or appropriated, by the child and become available on its psychological plane (following Vygotsky's well-known *genetic law of development*). The process of guidance involves building bridges from the child's present understanding and skills to the requirements of the new task, the selection and structuring of the activity, and the gradual transfer of responsibility from the caretaker to the child; guidance may be explicit or tacit, challenging or supportive (Rogoff 1990, p. 8).

The promise of scaffolding as an instructional approach is evidenced by the process of language acquisition of young children through verbal interaction with their caretakers sketched above. Similarly, apprentices accomplish the learning of skills in trades and crafts through participation in relevant activities guided by their masters (Rogoff 1990, pp. 90–91). Research on such traditional apprenticeships inspired influential new educational approaches such as the *cognitive apprenticeship* approach (Collins et al. 1989). Similar to traditional apprenticeship, cognitive apprenticeship emphasizes participation in real-world problems under the close attention of more experienced persons who provide support and guidance instantly. Yet, cognitive apprenticeship focuses on cognitive skills such as reading, writing, and mathematical problem solving (whereas traditional apprenticeships are often concerned with externally visible physical activities) and aims at generalized knowledge that can be used across a variety of situations (whereas such a level of flexibility is in traditional apprenticeships often not required and targeted). At its core, cognitive apprenticeship comprises the elements *modeling* (displaying and explaining expert problem solving), *coaching* (on-task support including scaffolding, feedback, and remodeling of selected task aspects), and *fading* (gradually reducing support and transfer responsibility). Sometimes, these elements are referred to as the *modeling-scaffolding-fading* tutoring strategy (Graesser et al. 2001). Other elements, probably less common in many traditional apprenticeships, include *articulation* (getting students to explicitly describe aspects of their knowledge, reasoning, and problem solving), *reflection* (getting students to compare their problem solving with the one of experts or among each other), and *exploration* (getting students to identify and formulate appropriate questions, problems, and goals themselves).

The significance of scaffolding gets also obvious from research on expert tutoring, which identifies scaffolding as a critical component of successful tutoring. Studies of tutorial dialogues show that scaffolding moves comprise a major portion of expert tutor moves (e.g., Chi et al. 2008 – 36%; D'Mello et al. 2010 – 47%; note that operational definitions of scaffolding may slightly differ). This is remarkable since "good" tutoring has been found to be far more effective than classroom instruction (Bloom 1984); the two-sigma advantage in effect size of human tutoring over classroom instruction is regularly cited as benchmark in research on the effectiveness of intelligent tutoring systems (e.g., Corbett et al. 1997). Another result of studies of expert tutoring indicates a relatively high proportion of explanation and lecture moves [e.g., Chi et al. (2008) – 23%; D'Mello et al. (2010) – 30%]. D'Mello et al. (2010) suggest that lecturing may establish the knowledge foundation for more advanced tutoring moves such as scaffolding or modeling. Contrary to this, Chi et al. (2001) found that when tutors intentionally suppressed explanation and feedback moves and used a higher proportion of scaffolding instead, the tutoring was equally effective in terms of learning. Interestingly, Chi et al. (2001) worked with novice teachers, who generally used a much lower percentage of scaffolding moves (5%) and many more explanation moves (53%). This result suggests that the extent of scaffolding is a strong indicator of tutoring expertise. Chi et al. (2008) further investigated whether tutor moves, student moves, or their interaction is predictive of learning outcomes. They found no evidence for the tutor move hypothesis—just by looking at the proportion of scaffolding moves, it was not possible to statistically explain learning gains. However, evidence was found that the proportion of substantive student moves was indeed predictive, and likewise were fruitful tutor-tutee interactions (e.g., tutor scaffolding followed by a substantive tutee contribution). Chi et al. (2008) propose that *self-constructions* on the part of the tutee and *co-constructions* between tutor and tutee are keys to learning success, with the latter of the two maybe even more effective. Self-constructions in dialogue may be considered as the social equivalents to *self-explanations* in individual learning, which are known to be effective for learning (Chi et al. 1989; Chi et al. 1994). Co-construction sequences are often opened or maintained through scaffolding moves (questions, prompts, hints), which implicitly or explicitly request a response from the tutee.[8]

Generally, scaffolding is not restricted to human support but can also be implemented in other ways (Tabak 2004). For instance, the work of Scardamalia and Bereiter (1983, cited in Pea 2004) provides an example for scaffolding that is based on physical note cards. These note cards contain sentence starters aimed at eliciting and promoting advanced writing strategies from learners. They may be provided as long as the learners are not able to apply advanced writing strategies on their own. Instructional scaffolding may also be implemented in computer-based environments—an important current research topic discussed in detail in Chapter 3. Of course, the way scaffolding is realized, whether humans are involved or not, may be an important factor for the achieved success. For instance, our "social brains" may be specifically tuned towards human support (Mercer 2013).

2.6.2 Cooperative learning: Learning in groups of peers

The second arrangement of social learning is the learning in groups of peers, which is typically referred to as *cooperative learning*.[9] From the beginning of the 1970s on, educational research became increasingly interested in cooperative learning as an instructional approach (Slavin 1996). While the majority of empirical results indicate that cooperative learning can produce learning outcomes superior to individual

[8] Chi (2009) uses a revised terminology in her *active-constructive-interactive* conceptual framework. For what is referred to in Chi et al. (2008) as a *co-construction* between a tutor and a tutee, Chi (2009) refers to as a *guided construction*. *Co-construction*, on the other hand, she now uses for a specific type of interaction in a joint dialogue between peers, e.g., when one student completes an idea a fellow student has started to produce.

[9] Some researchers draw a distinction between cooperative learning and collaborative learning (Dillenbourg 1999). On the one hand, cooperative learning may be associated with a more coarse-grained mode of interaction, following a division of work pattern, e.g., analyzing a given task, defining sub-problems, assigning sub-problems to different group members, solving sub-problems independently, and finally, merging the individual parts to an overall solution. On the other hand, collaborative learning may be associated with more fine-grained pattern of interaction, with students not only coordinating but also jointly engaging in the actual problem-solving process. Yet, this distinction is not consistently made in the literature and also not of further importance for the work presented here. Both terms are used interchangeably in this dissertation with a preference for the term used by the originator of the currently discussed piece of cited work.

learning arrangements, some studies could not confirm the superiority of cooperative learning, raising the question for the specific conditions for successful application (Dillenbourg et al. 1996). An influential account to explain effects of different cooperative learning arrangements (or their absence) is provided by the *social interdependence theory* (Johnson et al. 1981; Johnson and Johnson 2009). A basic tenet of this theory is that learning success can be explained by considering interdependencies between group members. Interdependencies may arise from different factors, most notably, the goals individual group members pursue, resources they can contribute to the task (e.g., knowledge or materials), and rewards that are called out for success. Some notable empirical results from longstanding research and meta-analysis of cooperative learning are (Johnson and Johnson 2009):

- *Positive goal interdependence* means that the attainment of each group member's goals is positively correlated with the goals of other group members (e.g., the group has to compose a joint report). Group members may be more inclined to help others when they know that their helping is conducive to their own goals as well. Empirical results identify goal interdependence as a crucial prerequisite for group productivity and learning.
- *Positive resource interdependence* means that to achieve their goals group members mutually depend on resources that only other group members can provide (e.g., each group member has unique knowledge necessary for successful task completion). Resource interdependence thus creates a necessity to interact with others, and prevents students from working separately. Yet, empirical results show that resource interdependence has negative effects when goal interdependence has not been established. Then, group members often focus on getting resources from others but are less inclined in sharing their own resources since their own goals do not benefit (or do even suffer) when other group members attain their goals.
- *Positive reward interdependence* means that possible rewards are made contingent on the performance of other group members (e.g., rewards may be determined based on the performance of the group as a whole). A reward may be a certificate, other kinds of public recognition, or a good grade. Empirical results suggest that reward interdependence can increase the positive effects of goal interdependence.

Slavin (1996) emphasizes the importance of group goals, group rewards, and individual accountability. The definition of group goals essentially leads to the condition of positive goal interdependence discussed above. Individual accountability means that group success is made contingent on each member's individual learning. Otherwise, more skilled group members may take over and solve the problem independently, without much involvement of the weaker ones, which may lead to a better task performance but less learning. One approach to setup such a goal and incentive structure is, for instance, to give grades based on the total of individual knowledge gains. According to Slavin (1996), the combination of group goals, individual accountability, and rewards increases students' motivation to (1) learn themselves, (2) encourage other group members to learn, and (3) help other group members to learn. This again can be expected to lead to actions and interactions conducive to learning, for instance, more intense cognitive elaboration of contents, more elaborated explanations, and mutual support.

Cohen (1994) makes a distinction between *true group tasks*, which essentially require the contribution of all group members to achieve a positive result, and other tasks that can principally also be solved in an individual effort (e.g., in collaborative seatwork). Another important factor is, according Cohen, whether problems are *ill-structured* or not, that is, whether "right" solutions and well-defined problem-solving procedures do exist. Typically, if the objective is to promote higher-level skills, ill-structured problems are used. Based on a comparison of own results and ones obtained by Webb (see, for instance, Webb 1989), Cohen concludes that there is a crucial difference between true group tasks involving ill-structured problems and the kind of tasks that are addressed in collaborative seatwork. True group tasks with ill-structured problems critically depend on an open and elaborated exchange between students, and certainly involve elements of argumentation. There is a reciprocal dependency between group members since the task requires everyone to both contribute and receive information and help. The overall frequency of task-related interactions has been found to be indicative of individual learning gains. The interaction patterns in more routine tasks that can also be tackled individually are typically of a simpler kind. The interdependence between group members is often unidirectional rather than reciprocal, that is, some students depend on other students but not vice versa. Cohen cites empirical results showing that learning gains could not be predicted from the overall frequency of interactions. Instead, patterns of help giving and help receiving seem to be important for learning. In particular, help giving in form of detailed, elaborative explanations has been identified as the most reliable predictor for learning. Help receiving has been found to be only effective under specific conditions: First, the help must match the information need of the addressee (e.g., errors and high-level questions of the addressee require better elaborated help). Second, the received help must be applied to the problem. Other results suggest that receiving appropriate help is more likely when help requests are more specific. Generally, argumentation and critical discussion play a minor part in such arrangements.

There are some interesting parallels to the research on peer tutoring discussed before (Chi et al. 2008). First, the interaction between learning partners seems to be important. In both cases, the learning of the supported party can be best explained by also considering the response of the supported party to the provided help or scaffolding. Second, there is a danger that learners help one another only to the extent that progress on the current task is made without providing sufficiently elaborated explanations to promote conceptual understanding. In both settings, unelaborated responses—feedback in the case of tutoring and help in the case of cooperative learning—were negatively related to the achievement of the addressee.

Collaborative seatwork groups are often composed to be heterogeneous with respect to student ability. The benefit of the social arrangement then may lie in that peer tutoring situations emerge quite naturally. As pointed out by Johnson and Johnson (2009), certain tasks may be better approached through individual learning, e.g., when the task is "unitary, nondivisible, and simple," focusing on "the learning of specific facts" or "simple skills." Such tasks may also benefit from a collaborative seatwork arrangement in which the interaction is often restricted to sporadic question-answer exchanges. If the focus is on argumentation skills and conceptual learning, however, richer forms of (e.g. argumentative) interactions, which are most likely to occur in true group tasks involving questions that are to some extent ill-defined, may be far more effective.

The work of Webb (1989), Cohen (1994), and other scholars marks a general change in the research paradigm of cooperative learning (Dillenbourg et al. 1996). Early research focused on comparisons between the effects of individual and cooperative/collaborative arrangements on learning with the result that collaboration was mostly, but not always, the superior approach (the *effect paradigm*). Researchers then became increasingly interested in the specific conditions under which collaborative arrangements actually lead to learning or fail (the *condition paradigm*). In this category falls the research on goal and incentive structures discussed above. Other conditions of interest were group composition (e.g., heterogeneous or homogeneous groups with respect to the general ability level), individual prerequisites (e.g., minimum development stage for engaging in productive collaboration), and task features and requirements (e.g., higher-level skills or more routine skills). Research based on this approach led to partly contradictory and puzzling results. Apparently, effects on learning often emerged from a complex interplay of different variables. For instance, Slavin (1996) admits that his recommendations with respect to cooperative learning—establishing group goals and individual accountability—may be of a lesser importance under certain conditions: (1) Members of voluntary groups may be intrinsically motivated since they see a direct personal benefit in collaboration, otherwise they would not decide to participate. They may show an active engagement in group work to increase their chances to be invited again to participate in the future. (2) Controversial tasks without a single right answer quite naturally involve overt forms of reasoning (explaining and arguing), which may be beneficial for learning in themselves. So there is less of a need to provide incentives for students to teach each other. (3) Highly structured (or scripted) tasks directly affect the interactions between students. Beneficial interaction patterns may then emerge from the given structuring of the task rather than from a motivating goal and incentive structure. The observation that conditions often only indirectly influence the learning success motivated the adoption of a new paradigm, the *interaction paradigm*. Essentially, the "big" question for the relationship between the nature of collaborative arrangements and learning effects is split into two more manageable sub-questions. The basic assumption is that the true value of collaborative learning lies in the interactions between learning partners. Therefore, *interaction* is introduced as a mediating variable between *condition* (e.g., different student populations or different instructional techniques) and resultant *learning outcome*. The first of the two smaller questions asks which interaction patterns are triggered under which conditions. The second question asks which interaction patterns are conducive to and which ones detrimental to learning. Figure 4 depicts the causal structure typically investigated in studies of collaborative learning. The first causal pathway assumes that conditions lead to interactions and interactions lead to learning. The second causal pathway represents the possibility that some learning may be directly attributed to the condition (e.g., if students do not engage in collaboration and solve a given problem in large parts independently, potential learning gains cannot be plausibly related to their interactions). One group of interactions considered particularly valuable is argumentative interactions. Correspondingly, instructional approaches in collaborative learning often try to foster specific forms of argumentative exchange.

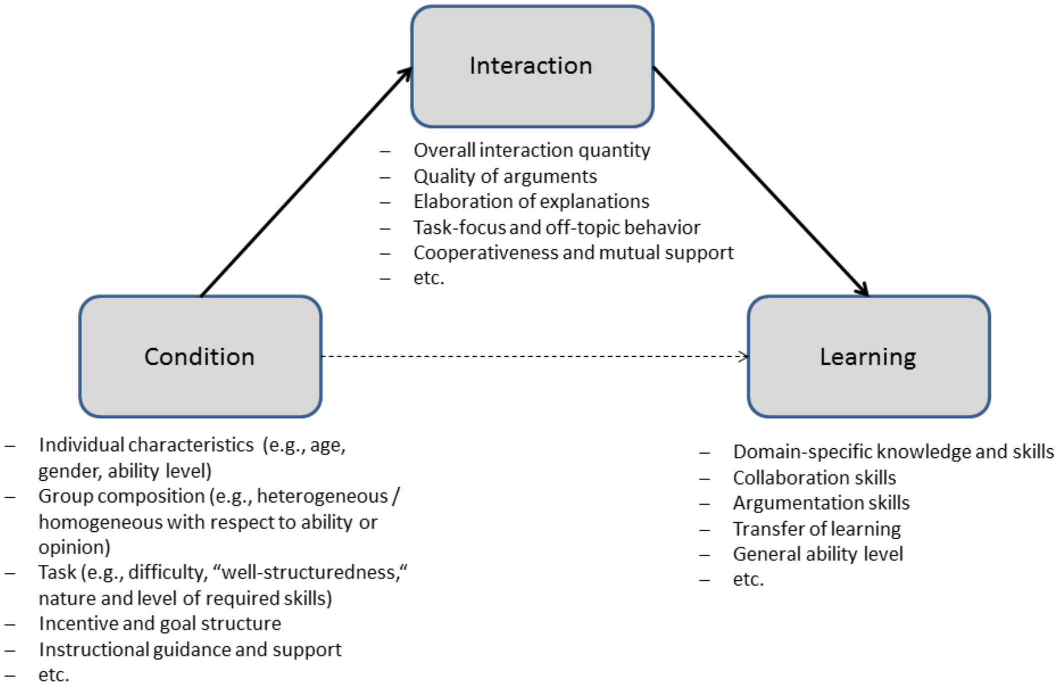

Interaction
– Overall interaction quantity
– Quality of arguments
– Elaboration of explanations
– Task-focus and off-topic behavior
– Cooperativeness and mutual support
– etc.
Condition
– Individual characteristics (e.g., age, gender, ability level)
– Group composition (e.g., heterogeneous / homogeneous with respect to ability or opinion)
– Task (e.g., difficulty, "well-structuredness," nature and level of required skills)
– Incentive and goal structure
– Instructional guidance and support
– etc.
Learning
– Domain-specific knowledge and skills
– Collaboration skills
– Argumentation skills
– Transfer of learning
– General ability level
– etc.

Figure 4: Causal structure typically investigated in collaborative learning research.

2.6.3 Mechanisms of learning in social arrangements

The discussion above of the two main social learning arrangements—learning with a more knowledgeable person and learning with peers of equal status—leads to a number of possible mechanisms of collaborative learning. Mercer (2013) proposes three hypotheses of how language-based collaborative activities may lead to individual learning: appropriation, co-construction, and transformation.

- *Appropriation:* Learners explain or exhibit their knowledge and problem-solving strategies to other students, who *construct* their own knowledge representations and problem-solving strategies from the perceived input. Hence, learning is explained through the individual processing of information *transmitted* between learners. Other group members are sources of information one may tap. Each interaction may be considered as unidirectional transfer of information, which triggers processes of individual cognitive processing. For instance, the knowledge *self-constructions* of tutees identified by Chi et al. (2008), which followed on a tutor's previous explanations, may be considered as the result of an appropriation in the Mercer sense. Anderson et al. (2001)

hypothesize that students pick up and appropriate argument stratagems (i.e., specific patterns of argumentation and corresponding language forms) they observe in discussions with other students. Generally, Mercer's definition of appropriation stands in line with more traditional, cognitive frameworks, which explain social learning from a perspective of individual cognitive processing of perceptions made in the social surrounding ("social-as-context view"; Suthers 2006). It differs in this respect from other uses of the term (e.g., Rogoff 1995). Two more specific theoretical notions based on the assumption of individual cognitive processing, socio-cognitive conflict and cognitive elaboration, will be discussed below.

– *Co-construction:* Collaborators jointly construct new knowledge or devise new strategies that could be used subsequently in individual situations (or new collaborative situations). The new knowledge or strategies are genuinely born out of reciprocal interactions of collaborating learning partners. Mercer (2013) cites some interesting phenomena found in empirical studies supporting the co-construction hypothesis. For instance, some research showed that social sensitivity of group members and evenness of participation can be stronger indicators of group success than the group members' average intelligence. That is, group-level processes can make a distinctive contribution, providing advantages over purely individual processes. Examples of co-constructed knowledge are co-elaborated conceptions of good argumentation (Kuhn and Udell 2003), jointly developed discussion ground rules (Wegerif et al. 1999), and co-constructed arguments (Kuhn et al. 1997; Resnick et al. 2010). The concept of transactive discussion moves, which involve "reasoning on the reasoning of others" (Berkowitz and Gibbs 1985), quite naturally matches with the notion of co-construction. As pointed out by Mercer (2013), also the more knowledgeable part in a scaffolding arrangement may gain from co-construction activities in that his original conceptions may be sharpened, become more explicit and clearer through co-constructed representations of knowledge and strategies. Moreover, the awareness of possible own gaps in understanding may increase. Overall, the co-construction hypothesis assigns a more crucial role to *inter*-mental processes, which reflect joint cognitive efforts. To account for processes of co-construction, one may employ group-level constructs such as *shared knowledge*, *joint problem space*, or *collective reasoning*. Taken to the extreme, the collection of individual minds may be considered as a single cognitive system (Salomon 1997; Stahl 2006).

- *Transformation:* Collective and scaffolded reasoning may have transformative effects on individual reasoning, that is, rather than acquiring specific knowledge and skills, discussants may change their mode of thinking in more fundamental ways. This hypothesized mechanism is consistent with Vygotsky's (1978) view on human development, expressed in his genetic law of development, which postulates that all higher-level mental functions first appear on the social plane, are gradually internalized, and become increasingly available on the psychological plane. The idea that multi-perspective forms of thinking are internalized versions of experienced group dialogues falls into this category (Wegerif et al. 1999). Similarly, the development of self-regulation competency, which is an important factor for learning success, may have its origins in social interactions (Mercer 2013). For instance, young children self-regulate activities by talking to themselves. It has been observed that children's self-talk resembles in form and function the speech adults previously used to scaffold exactly these activities. A possible explanation is that self-talk constitutes one stage in the process of internalizing socially experienced forms of regulation. The pattern of self-regulation through speaking aloud ceases at later ages; self-regulation is now accomplished through an inner voice.

As pointed out by Mercer, the three hypotheses are not mutually exclusive, that is, all mechanisms might contribute to the learning in social contexts. In fact, a process of appropriation may be conceived of to explain how jointly developed knowledge (in the case of co-construction) or socially experienced forms of reasoning and regulation (in the case of transformation) become part of a learner's internal repertoire. The definition of appropriation proposed by Mercer (2013) assumes that information is transmitted between learning partners—typically through the medium of language—and then mentally processed by the receivers. Two theoretical frameworks that give a more detailed account of how knowledge emerges and develops through individual cognitive processing are the *cognitive elaboration theory* and the *socio-cognitive conflict theory*. These two widely adopted theories are particularly relevant with respect to argumentation learning (Nussbaum 2008):

- *Cognitive elaboration:* Collaborative learning activities may trigger processes of deep cognitive processing and knowledge elaboration. Producing, interpreting, and evaluating arguments and explanations require an active processing of information at a deep level. For instance, when evaluating arguments in terms of plausibility and evidence, the presented assertions must be put into relation with existing knowledge structures. When developing explanations and arguments, connections between knowledge elements may be strengthened or created; new knowledge elements may be inferred; own misunderstandings and knowledge gaps may be detected and repaired; clearer and more precise language-based representations may be developed. Empirical results in research on helping behavior in cooperative learning groups (e.g., Webb 1989) indicates that constructing explanations is often more effective than receiving explanations. In line with the cognitive elaboration hypothesis, similar cognitive elaboration processes may be assumed as for the self-explanation effect (Chi et al. 2001). Based on a meta-analysis of tutor learning in peer tutoring settings, Roscoe and Chi (2007) conclude that the benefit for tutors lies in developing, rather than just delivering, explanations, emphasizing the importance of active cognitive engagement.

- *Socio-cognitive conflict*: This theory builds upon Piaget's constructivism and his notion of equilibration, which describes the relation between mental state and physical environment in terms of two complementary mental processes (Piaget 1985). *Assimilation* means that existing mental structures, or schemata, are used to interpret and make sense of sensory input perceived from the external world. *Accommodation* means that mental structures are modified and brought into accordance with the perceived sensory inputs. Cognitive development is the result of the continuous attempt to resolve conflicts between the perceived outside world and mental representations through processes of assimilation and accommodation in order to achieve a state of equilibrium. A basic assumption of proponents of the socio-cognitive conflict theory is that conflicts that come to light in social situations play a particular role in cognitive development. Such conflicts create a double impetus for resolution since they have an intrapersonal and interpersonal dimension. In a series of experiments, predictions of the socio-cognitive conflict theory were confirmed (Doise and Mugny 1984). For instance, pairing children with different (or, conflicting) cognitive strategies led to more learning than pairing children with identical strategies (Mugny and Doise 1978).

  A well-established result is that not all conflicts are conducive to learning—conflicts should be addressed in constructive ways. Research on conflict resolution (Deutsch 2005) and classroom group learning (Johnson and Johnson 2009) emphasizes the superiority of cooperative over competitive approaches to conflict resolution, both in terms of processes and outcomes. Conditions of social interdependence found to be essential for cooperative learning (e.g., positive goal interdependence; see discussion above) can be expected to promote exactly these kinds of interactions. In a similar vein, empirical results

> indicate better learning results when *epistemic* rather than *relational* conflict resolution strategies are used (Darnon et al. 2007). Epistemic resolution strategies focus on the task and aim at identifying the best possible solution through reasoned argument. Relational resolution strategies aim at defending one's social status and power, which are at risk when others appear to be more competent and knowledgeable. So while epistemic approaches are cooperative in nature, relational approaches can be characterized as competitive. Whether learners perceive the interaction as relational or epistemic essentially depends on the design of the instructional situation (Buchs et al. 2010).

A unifying conceptual framework from the cognitive elaboration tradition, linking learning activities, cognitive processes, and learning outcomes, is provided with the *active-constructive-interactive framework* (Chi 2009; later referred to as *Differentiated Overt Learning Activities* [DOLA; Menekse et al. 2013]). The framework explains results from both, social and individual learning research. It groups overt learning activities into four broad classes (*passive*, *active*, *constructive*, and *interactive* learning activities) and proposes potential cognitive processes at work for each. *Active learning activities* are those that engage the learner in some form or other, for instance, underlining or paraphrasing some passage of text. Such activities are assumed to trigger cognitive processes such as activating existing knowledge, or assimilating and storing new knowledge. *Constructive learning activities* involve creating ideas not explicitly contained in provided learning materials. For instance, self-explaining typically requires filling in information not explicitly mentioned. Similarly, creating concept maps often requires identifying connections between concepts not explicitly stated. Such activities potentially trigger cognitive processes that go beyond the ones postulated for active learning activities. Rather than just adding and assimilating new knowledge, more far reaching inferences and connections may be made; bugs and gaps in the own understanding may be detected; existing knowledge structures may be reorganized or repaired. Finally, *interactive learning activities* are most typically seen in dialogues between human actors. In instructional dialogues, tutors typically provide scaffolding to guide tutees in producing meaningful responses (*guided construction*). In peer dialogues, students may, for instance, elaborate on or argue against contributions of fellow students (*sequential-construction* or *co-construction,* depending on the granularity of interaction). Interactive learning activities are assumed to essentially trigger the same cognitive processes as constructive ones. Yet, additional advantages emerge from the more dynamic process these activities are embedded in. In particular, learning partners (or tutor and tutee) mutually influence one another by contributing information, arguments, questions, feedback, scaffolding, etc. to the discourse. This happens immediately and in a highly adaptive fashion, tailored to the requirements of the situation, the course of the discourse, and the participants' goals. Moreover, as a result of sequential construction and co-construction, ideas that students are not able to produce alone may be produced in a joint effort, leading to a deeper or novel understanding. Chi (2009) formulates the following hypothesis regarding the learning effectiveness of the different activity classes: interactive > constructive > active > passive (*ICAP hypothesis*; Menekse et al. [2013]). While Chi (2009) discusses existing empirical studies in line with the ICAP hypothesis, Menekse et al. (2013) present positive evidence from two studies explicitly designed to test the ICAP hypothesis.

2.7 Argumentative Learning in Social Arrangements

This section focusses specifically on the role of argumentation in social learning arrangements. The first subsection briefly describes, in general terms, how argumentation can be employed and fostered in social learning arrangements. The following subsections review empirical results of educational studies grouped according to the pedagogical aims of the intervention. First, interventions may target the learning of argumentation skills themselves. Second, in line with the interaction paradigm discussed above, argumentative interactions may be seen as a particularly useful way to elaborate and learn subject matter contents at a deep level. Third, engagement in argumentative activities may be seen as a way to improve thinking skills more generally. Sociocultural theorists see the origin of thinking in participation in social exchange and discussion. Accordingly, critical thinking skills may be learned through participation in argumentative activities. Even if not following the sociocultural school of thinking, it must be acknowledged that argumentation is a basic skill important across academic disciplines, professions, and everyday life contexts. The learning of patterns of argumentation may thus indeed be conducive to thinking and problem-solving in a wide variety of situations within and across domains.

2.7.1 Social learning approaches to argumentation

Research on traditional scaffolding approaches in form of one-to-one learning sessions is relatively scarce. In reference to other work, Kuhn (1991) mentions the possibility that scaffolding may be realized in one-to-one tutoring sessions between a learner and an instructor who provides prompts and hints to support the learner in producing better arguments. A possible reason why such research is less common is certainly that one-to-one tutoring has a high demand on qualified human tutors, which makes the approach less attractive from a practical point of view. The preferred social arrangement with respect to argumentation learning is therefore the learning in groups of peers. Overall, two main approaches can be distinguished.

First, important concepts and principles of argumentation may be co-elaborated in group discussions rather than defined and presented by the teacher. For instance, group discussions may revolve around the question of what makes a good reason, different types of evidence and their force (Kuhn and Udell 2003), or ground rules for productive discussions (Wegerif et al. 1999). The active involvement of students is in line with constructivist learning theories and from this perspective preferable over a didactic presentation of contents. Moreover, when developing discussion rules jointly in the group, students are more likely to accept ownership of and commitment to these rules (Wegerif et al. 1999).

Second, group-based activities can also take the form of engagement in argumentative discourse. Certain argumentative behaviors may be best invoked in interaction with critical opponents who are actually present rather than just anticipated. These critical opponents essentially serve a twofold role. On the one hand, they act as evaluators and critics of one's own position. On the other hand, they present claims and arguments one has to evaluate and critique. Thus, the presence of critical opponents quite naturally generates manifold opportunities to practice argumentation skills in a realistic situation. Another possible learning mechanism is that discussants may take the behavior of others as a model for the construction of their own discussion moves. Based on this idea, Anderson et al. (2001) hypothesize that discussion moves that are functional (e.g., in terms of their persuasive force) and at the same time not overly complicated may spread in discussions from their originators to other fellow discussants (*snowball hypothesis*).

However, research shows that learning in groups does not unconditionally lead to positive learning outcomes (Cohen 1994). Learners often lack essential social and cognitive skills and strategies to engage in productive collaboration and discussion. Some of the approaches mentioned above therefore involve coaches who support group discussions (Kuhn and Udell 2003; Wegerif et al. 1999), which can be understood as scaffolding at the group level.

Since the beginning of the 1980s, another approach, called *scripted cooperation*, became increasingly popular to provide structure and guidance to cooperation processes (O'Donnell and Dansereau 1992). The notion of a *cooperation script* is inspired by scripts in the theater, which define the actors' roles and the nature and timing of their activities. In a similar fashion, student roles may be *scripted* (e.g., a critic and a proponent of a position), and likewise specific learning activities, their sequencing, and timing (e.g., first: student-1 presents argument, then: student-2 proposes counterargument, then: student-1 rebuts counterargument, go to step 1 again and repeat with switched roles). In this way, interactions assumed to be beneficial for learning may be invoked in a more controlled and targeted fashion. Moreover, the practiced patterns of interaction themselves (e.g., specific argumentation moves or sequences) may be internalized and become available on future occasions. A particularly promising approach to scripting is the use of computer technology, which allows the execution of cooperation scripts in a repeatable and automated manner, while at the same time recording traces of student and system behavior for later analysis. The overall concept of computer-based cooperation scripts, which is of special importance for the presented work, will be discussed in greater detail in section 3.3 along with specific scripting approaches.

2.7.2 Learning to argue

The research team of Deanna Kuhn conducted a series of studies (Kuhn et al. 1997; Kuhn and Udell 2003) to investigate whether argumentation skills can be improved through engagement in dialogical argumentation activities (i.e., activities that explicitly address claims and arguments of an opposing party). The interventions and tests of the studies reported below focused on the controversial topic of capital punishment. Relatively weak comparison conditions were used, which either did not include any learning activity as replacement for the experimental intervention (Kuhn et al. 1997), or which only involved a subset of the activities of the treatment condition, resulting in considerably less learning time (Kuhn and Udell 2003). Thus, the following findings do not allow conclusions on the question whether the specific treatments are superior to alternative ones (e.g., reading of a text book on proper argumentation). Rather, these studies show that dialogical argumentation activities can principally be used to foster the learning of argumentation skills.

Kuhn et al. (1997) investigated the effects discussions have on argumentation skills. Participants in a treatment condition met five times over a period of five weeks—each time with a different partner—to discuss the topic of capital punishment. Participants in a control condition only took the pretest and the posttest without any additional learning activity in between. Pretest and posttest requested from participants their opinion regarding capital punishment and an argument to justify this opinion. The analysis showed that the discussions had a positive impact on the quality of reasoning. Particularly frequent types of improvements from pretest to posttest were a move from one-sided arguments to two-sided arguments (i.e., arguments containing both pro and con aspects regarding one's own position) and the use of comparative arguments (i.e., arguments cast within a framework of multiple alternatives). An analysis of the discussion transcripts shed some light on mechanisms that may explain these positive effects. All argument elements that were newly introduced in the posttest in fact originated from the discussions. Many of these elements were not simply transmitted from one partner to the other. Rather, these elements were newly developed in response to the partner, or even co-developed by both partners over multiple turns. Possible changes of opinion between pretest and posttest were classified in four categories: *articulation* (from a neutral or near-neutral position to a moderate position), *polarization* (from a moderate to an extreme position), *centration* (from a more to a less extreme position), and *side change* (change from a pro to a con position or vice versa) The results showed that such opinion changes happened far more frequently in the treatment condition, that is, the discussions also had an impact on the participants' opinion formation (even if self-reports indicated that participants were themselves not aware of the extent of their opinion change). In a second study, participants in the treatment condition engaged in a discussion with a partner only once. The results did not yield a significant difference to the control condition, highlighting the importance of sustained engagement in discussion practice and the potential value of being confronted with multiple different views.

Kuhn and Udell (2003) investigated the effects of a teaching approach that explicitly addresses the dialogical dimension of argumentation, i.e., counter-positions, counterarguments, rebuttals, etc. Teams were composed with four to eight students sharing the same opinion regarding the topic of capital punishment (pro or con). Control group teams only participated in the first half of an activity sequence (seven 90-minute sessions), which focused on the development of an argument for one's own position. The specific activities included: generating reasons, elaborating reasons, supporting reasons with evidence, evaluating reasons, and developing reasons into an argument. Treatment group teams in addition participated in another nine sessions, in which a debate with an opposing team was prepared and conducted. The specific activities included: examining and evaluating reasons of the opposite side, generating counterarguments against reasons of the opposite side, generating rebuttals against possible counterarguments of the opposite side, contemplating mixed evidence, and conducting and evaluating two-sided arguments. In both conditions, an adult coach provided instructions, explanations and guidance throughout the different activities (*social scaffolding*). As part of the activities, external representations of ideas in form of cards were employed with different colors indicating categories such as reason, counterargument, and rebuttal *(representational scaffolding*). To measure changes in discussion behavior, pairs of students engaged before and after the intervention in a discussion about capital punishment. The results showed that by addressing dialogical aspects in an explicit way, the use of *challenge moves* increased (identifying and challenging weaknesses of the counter-position) and the use of *exposition moves* decreased (i.e., presentation and clarification of one's own perspective). Positive treatment effects were also visible in the arguments composed in a pretest and a posttest. Qualitative improvements occurred significantly more often in the treatment condition. Moreover, treatment students often dropped weaker reasons they used before in the pretest and included more powerful new reasons instead, a pattern not present in the control condition. That is, participants did learn a broader range of arguments and strategically selected the ones they evaluated most effective in supporting their position. Even if not explicitly tested, Kuhn and Udell (2003) see the used social and representational scaffolds as major contributors to the achieved improvements.

Reznitskaya et al. (2009) developed a pedagogical approach called *collaborative reasoning* aiming at promoting pedagogically valuable forms of discussion in elementary school classrooms. Collaborative reasoning involves a free and open exchange between students about a given "big" question, which is a question without definitely right or wrong answers. The big questions are motivated by a story students read before entering the discussion. The discussions are argumentative in nature and managed by the students themselves. Collaborative reasoning is contrasted with recitations, a teacher-centered discussion format which is still prevalent in classrooms. In recitations, the teacher typically asks a series of questions focusing on text understanding and related general knowledge, and nominates students who raise their hands to respond to these questions. The flow of the discussion is controlled and mediated by the teacher. Due to the nature of the overall setup, the discussion does entail no or little argumentation; often, the discussion takes the form of a sequence of questions and answers. To establish collaborative reasoning in the classroom, teachers attend a one-day workshop and are coached by members of the research team based on in-class observations. The facilitation strategy to promote collaborative reasoning includes, among other things: prompting students to express their positions and reasons, challenging students' reasoning, highlighting effective use of argument stratagems (i.e., language patterns reflecting desired forms of reasoning), and modeling reasoning by thinking aloud. Following the idea of scaffolding, teachers gradually transfer responsibilities for discussion management to students.

A number of empirical studies have been conducted to analyze various aspects of collaborative reasoning. The studies involved up to ten collaborative reasoning sessions. Studies focusing on interactions revealed that, compared to recitations, collaborative reasoning discussions: (a) contained significantly more peer interactions, (b) focused more on underlying moral and social questions rather than specific story contents, and (c) contained more dialogic interaction and reasoned judgments. In support of the *snowball hypothesis* (Anderson et al. 2001, see above), it was found that when a child successfully used an argument stratagem, other children often used the stratagem in subsequent situations as well. Studies focusing on the effect on individual argumentation skills found significant differences and large effect sizes in terms of satisfactory arguments, counterarguments, and rebuttals. (Individual argumentation skills were measured through analysis of reflective essays written after the intervention and addressing a new story and moral dilemma.) Studies focusing on the relation between collaborative reasoning interactions and individual achievement found that (a) the number of reasons a student produces during group discussions predicts the number of reasons he produces in an individual essay, and (b) the number of counterarguments discussed in a group predicts the number of counterarguments group members consider in their individual essays. These results suggest that the active production of reasons and the confrontation with opposing arguments fosters the development of reasoned, multi-perspective forms of thinking.

Zohar and Nemet (2002) developed an instructional unit (about twelve lessons) that integrates the teaching of argumentation skills and biology content. The learning materials consisted of practical activities (based on ten moral dilemmas about human genetics) and explicit instructions on biology concepts and argumentation. The dilemmas were chosen to provide authentic, interesting, and relevant problems suitable to stimulate patterns of scientific argumentation. Practical activities included writing assignments and discussions. In a study, students in the treatment condition were taught by the instructional unit sketched above while students in a control condition where taught by traditional methods (textbook, presentation of contents by teacher, standard genetic problems as exercises). Domain and argumentation learning was assessed based on a pretest and a posttest. The results showed significant advantages for the treatment condition both in terms of biology knowledge and argumentation skills. Notably, not only argumentation about the covered genetic contents improved but also argumentation about everyday issues. While the improvements in some of the previously discussed studies (Kuhn et al. 1997; Kuhn and Udell 2003) may be explained by an increased understanding of domain knowledge, the Zohar and Nemet (2002) result thus clearly shows that domain-general argumentation skills can be learned. Osborne et al. (2004) consider the twelve lessons of engagement in argumentation activities in the Zohar and Nemet (2002) study as generally too limited to achieve clear improvements. In an own study, they tested a nine-month intervention program in which argumentation activities were integrated into a science course. Their results were overall positive but did not reach a significant level. A second important observation of Zohar and Nemet (2002) is that argumentation activities apparently promoted the learning of domain knowledge, an aspect addressed in the next section.

2.7.3 Arguing to learn

The previous section already hinted at the double role argumentation can play in cooperative learning settings. On the one hand, argumentation skills can be an end in itself, that is, participation in argumentative activities is used as an instructional approach to foster argumentative reasoning skills (*learning to argue*). On the other hand, argumentative activities can be a means to learn specific subject matters (*arguing to learn*). Both pedagogical objectives are not mutually exclusive and typically pursued in parallel. Clear evidence for the effectiveness of argumentation for the learning of subject matter was found in the Zohar and Nemet (2002) work. The studies of Kuhn and colleagues (Kuhn et al. 1997; Kuhn and Udell 2003) showed that participation in argumentation-based activities helped students to improve their ability to produce arguments of higher quality. Since argumentation skills were assessed based on the same issue as was used during the treatment (capital punishment), it is at least plausible to assume that benefits stem from both, improved argumentation skills and content knowledge.

Resnick et al. (2010) give an overview of *Accountable Talk*, an approach to establish the norms of reasoned dialogue in the classroom. The goal is to go beyond the teaching of authoritative knowledge and procedural skills by targeting “learning with understanding.” The verbalization of disciplinary reasoning is an important component in this endeavor. Accountable Talk has its origins in research on classroom discussions. It builds upon the sociocultural notion that individual abilities and dispositions develop from participation in corresponding social practices. Accountable Talk has three components. *Accountability to the learning community* addresses the social dimension. It involves aspects such as listening to others, building upon the ideas of others, asking questions for clarification, and justifying agreement and disagreement. *Accountability to the standards of reasoning* addresses the logical dimension. It involves the making of sound inferences and the provision of reasonable explanations. Finally, *accountability to knowledge* is about basing one's claims and explanations on facts, published texts, and other kinds of available information, and using the appropriate vocabulary and language. Authoritative knowledge is still considered important since proper reasoning and discourse depend on a well-developed and accurate knowledge base. The role of the teacher is to structure and guide classroom discussions. Accountable Talk defines specific moves teachers can use to stimulate the desired forms of discussion. For instance, *revoicing* means that the teacher provides alternative formulations of what students said before (e.g., "So let me see if I've got your idea right. Are you saying …?"). Through revoicing a teacher can model how to express important ideas in an expert-like, genre-specific language. Other moves include asking students to restate one another's reasoning (e.g., "Can you repeat …?"), to give a reasoned opinion regarding something said before (e.g., "Do you agree or disagree and why?"), to build on what was said before (e.g., "Would someone like to add on?"), and challenges or counterexamples (e.g., "Is this always true?"). To put Accountable Talk into practice, teacher training materials are made available (Michaels et al. 2002), which are now being used in several large urban districts across the United States. Empirical evidence suggesting the high potential of Accountable Talk was gathered, for instance, in a four-year intervention program to foster talented students ("Project Challenge"). In a daily one-hour class, underprivileged children engaged in demanding math problems and teacher-led discussions that involved Accountable Talk moves. Tests conducted at several stages during and after the intervention showed impressive improvements in math-related skills, such as computation, mathematical understanding, and problem-solving. While a number of different factors may play a role in such broadly conceived, long-term interventions, the central role that discourse activities took makes it highly plausible that they made a significant contribution to the achieved success.

Results from a more controlled investigation in the domain of mathematics are discussed by Schwarz et al. (2000). They investigated whether, how, and under which conditions the dyadic interaction between peers with conceptual misunderstandings can lead to the development of a correct conception (the *two wrong can make a right* hypothesis). The investigation included a small-scale study, a main study, and case studies. It was conducted with underperforming high school students and covered the topic of decimal fractions. Based on pre-test results, students were classified according to whether they have a correct understanding or one of three different kinds of misconceptions. The small-scale study used three conditions formed based on a systematic pairing of students: (1) pairs of students with the same misconception (*wrong1-wrong1*), (2) pairs of students with different misconceptions (*wrong1-wrong2*), and (3) pairs consisting of one student with a correct understanding and another student with a misconception (*right-wrong*). Descriptive statistics suggested that only the interaction in wrong1-wrong2 dyads led to improvements. This observation was confirmed in the main study. A comparison between right-wrong dyads and wrong1-wrong2 dyads yielded a statistically significant difference. The success of the pairing of students with different misconceptions may be explained based on the socio-cognitive conflict and the argumentative interactions that emerged from diverging solution strategies. Such argumentative exchanges may occur in right-wrong dyads to a lesser extent since the *right* student may be confident in his approach and therefore not see much value in engaging in an argument with a less competent student.[10] The case studies yielded support for the hypothesis that argumentation is the driving factor. The analysis showed that conflicting problem-solving strategies led to argumentative sequences, including moves such as arguments, counterarguments, challenges, and concessions. Some patterns in the interactions suggest processes of co-construction and appropriation being at work (e.g., new insights gradually developed from the interaction with the partner; reuse of socially experienced reasoning patterns at later occasions). Another interesting finding is that students with certain misconceptions benefited more than other students. Apparently, students can only overcome a misconception when the content addressed in the argumentative exchange helps recognizing and repairing this misconception. The specific task design may have provided more opportunities to elaborate certain misconceptions than others.

Asterhan and Schwarz (2007) discuss two empirical studies, which investigated the effect of argumentation on conceptual learning in evolutionary biology with a population of undergraduate students. The results of the first study indicate the particular importance of argumentation activities for collaborative learning. In one condition student dyads were instructed to collaborate on a given task. In a second condition, student dyads were instructed to engage in argumentation with the goal to reach the best possible solution together. More specifically, the instructions included prompts to argue for and against each position, to provide adequate justifications, to provide evidence for claims, to identify weaknesses in arguments, and to check the

[10] Note that the situation here differs from expert tutoring (Chi et al. 2008) and collaboration under the condition of goal interdependence and individual accountability (Slavin 1996), both of which involve particular goal and incentive structures.

relevance of arguments with respect to their conclusions. In addition, a short excerpt of a fictional argumentative exchange was provided as an example. Conceptual understanding was measured in a pretest, a posttest, and a delayed posttest (one week after the intervention) in which given cases had to be analyzed in terms of Darwinist principles. The results were analyzed in two different ways. First, specific Darwinist principles were identified and rated (*fully correctly applied*, *partly correctly applied*, or *not mentioned*). Second, explanatory schemes of different quality were assigned to the answers (from *non-answers* to explanations that fully satisfy the Darwinist perspective). The results showed that both conditions improved from pretest to posttest but only the argumentation condition could maintain their gains over a longer period of time, as evidenced in the delayed posttest. The interaction transcripts were coded for quality of argumentative exchange from *no argumentation*, to *one-sided argumentation* (only one solution considered with justifications and explanations), to *dialectical argumentation* (considering two alternative solutions, or considering pros and cons of one position). The analysis of interaction type by condition showed that without specific instructional prompting, students are unlikely to engage in argumentation. The analysis of learning gains by interaction type showed that it is dialectical argumentation that is critical for maintaining conceptual knowledge gains over time. The second study investigated the role of argumentation in a non-collaborative context. In particular, the interactions within student dyads were restricted to a question-answer format. Study participants worked together with a *confederate* (i.e., a member of the research team who pretended to be a real student). In an experimental condition, the two dyad members were assigned different roles. The random assignment of roles to students was faked for the purposes of the study. The confederate was always the question asker and the real student was the question answerer. The confederate asked questions to engage the student in argumentative reasoning, e.g., to discuss strengths and weaknesses of their own and the (faked) confederate's solution. In a control condition, confederate and real student only read aloud their solutions to the other dyad member. The study result showed that the argumentation condition showed learning benefits in the posttest and could maintain these benefits in the delayed posttest. The control condition did neither show improvements in the posttest nor in the delayed posttest. An interesting observation in both studies is that the benefits of argumentation activities were mainly found with respect to the general explanatory schemes and not so much with respect to the mentioning of isolated principles. This observation suggests that argumentation is particularly conducive to deeper levels of learning and only to a lesser extent for the remembering of isolated propositions.

Asterhan and Schwarz (2009) present a reanalysis of data gathered in the Asterhan and Schwarz (2007) study. Based on the observation that the instructions to engage in argumentation were not always successful, they reexamined the data of the experimental condition of the first study (i.e., students prompted to engage in argumentation). In particular, student interactions were analyzed in greater detail and related to conceptual learning gains. On a macro-level, two factors have been identified to be important that at least one student improves considerably: (1) each student can be associated with a different explanatory schema [similar to the Schwarz et al. (2000) result] and (2) argumentation is dialectical rather than one-sided [in line with the Asterhan and Schwarz (2007) analysis]. On the micro-level, gaining dyads differed from non-gaining dyads in terms of dialectical moves (challenges, rebuttals, concessions, and oppositions) but not in terms of consensual moves (supports, agreements, and elaborations). These results are mirrored in an analysis of the correlation between individual students' gains, their own discussion moves, and their partners' discussion moves (in the subsample of wrong-wrong students). Again, dialectical moves came out as the key factor to learning, in particular for the student who produces dialectical moves. The most important individual type of move was the rebuttal, which reflects a complex interaction sequence comprised of a claim, a challenge, and the rebuttal itself. A crucial observation is that consensual moves, which aim at developing and verifying explanations, did not result in major conceptual improvements. This result seems to contradict results from research on expert tutoring (e.g., Roscoe and Chi 2007) and collaborative learning (e.g., Webb 1989), which highlight the positive impact explanatory activities can have on learning success. According to Asterhan and Schwarz (2009), evolutionary biology, which is the specific knowledge domain investigated in their study, is known for its particularly robust and persistent misconceptions. This may require *radical conceptual change* (i.e., restructuring knowledge fundamentally) rather than *incremental conceptual change* (i.e., adding or repairing individual pieces of knowledge). Radical conceptual change again may be more likely to follow from dialectical argumentation than from explanatory activities. Since the number of consensual moves did not differ between gaining and non-gaining dyads, it may actually be the combination of both types of moves—explanation and critical argument—that is most effective for learning.

2.7.4 Arguing to improve thinking

Some researchers even propose that participation in dialogical activities may be essential in the development of thinking skills more generally (Anderson et al. 2001; Wegerif et al. 1999). Dialogical accounts of thinking make a distinction between *dialogical reasoning*, which can be understood as a discussion between different perspectives and voices in one's head, and *monological reasoning*, which is lacking the multi-perspective view and thus constitutes a poorer form of reasoning. Neo-Vygotskian accounts of human development emphasize the role of participation in social practices as the driving force in the development of individual abilities, following the internalization / appropriation mechanism mentioned above. The combination of both theoretical stances suggests that participation in dialogical group reasoning is an important factor in the development of individual dialogical reasoning skills, which constitute the higher levels of thinking. To be effective in fostering dialogical reasoning skills, discussions themselves must embrace a "free and open encounter between different perspectives and ideas" (Wegerif et al. 1999). To furnish discussions with such an orientation, the teaching of corresponding discussion ground rules may be essential.

Wegerif et al. (1999) developed an intervention program (nine 60-minute sessions) to teach ground rules for discussions based on the notion of dialogical reasoning. The specific style of discussion fostered through these rules goes under the label of *exploratory talk*, and involves explicit and accountable reasoning (sharing relevant information, providing reasons for claims and criticisms), openness to challenges, exploration of alternatives, mutual encouragement to speak, and joint and consensual decision making. Activities within the intervention program included the modeling of desired behavior by the teacher, practicing of these behaviors by the students, joint elaboration of concrete discussion ground rules, and coached and unsupervised group discussions. In a study, the effect of the intervention program was evaluated against a control group who was not coached in exploratory talk. Before and after the intervention, students solved nonverbal puzzle problems indicative of general reasoning abilities, first in a small group and then individually. Group interactions were evaluated based on counting language patterns that are associated with exploratory talk (e.g., the use of the words *because* and the contribution lengths as indicators for exploratory talk). The analysis showed that the coaching of exploratory talk was effective in promoting the targeted patterns of language use and group reasoning. Individual test scores showed that the intervention was also effective in terms of individual reasoning improvements. This is remarkable since the intervention itself only involved coaching of discussion ground rules and did not relate in any way to the specific task of the performance test. Two alternative explanations may be employed. The weaker hypothesis is based on the observation that the individual performance task was preceded by the group performance task. This hypothesis states that exploratory talk helped students during the group performance test to jointly develop more effective reasoning strategies to solve the puzzle problem [in line with the mechanisms of appropriation and co-construction proposed by Mercer's (2013); see Table 17, first row]. The stronger hypothesis states that the learning of improved dialogical reasoning abilities had an immediate bearing on individual problem solving performance [in line with the mechanisms of transformation proposed by Mercer's (2013); see Table 17, second row). In any case, the intervention program had positive learning effects on group reasoning and direct or indirect bearing on individual problem solving.

More recently, Wegerif and colleagues within the Argunaut project (which parts of this dissertation originated from) expanded their interests from critical to creative group reasoning (Wegerif et al. 2010). In continuation of the lines of thinking sketched above, the presented research employs a collective rather than an individualistic conception of creativity. While critical reasoning can be associated with *deepening* the discussion, which often serves a filter function, e.g., by questioning implicit assumptions of claims and arguments, creative reasoning may be characterized by a *widening* of the discussion, manifested in new, emerging perspectives. Creativity is interpreted as a dialogical process, which, in "a dance of voices and perspectives," enhances understanding by creating new ways of seeing a problem. Wegerif et al. (2010) developed a coding scheme to manually label instances of creative reasoning (i.e., new perspectives) in graphical discussions (i.e., discussions represented as node-and-link graphs in which nodes represent contributions and links represent reply relations between contributions). A novel artificial intelligence approach was developed (*DOCE*; described in detail in McLaren et al. 2010) to automate the identification of deepening and widening moves. First validation results indicate a reasonable performance of the induced classifiers, suggesting that creative discussion moves can be reliably identified. Other results, based on qualitative research methods, suggest that new perspectives may be promoted through the spatial representation of the discussion in a graph, which gives a better overview of present themes and ideas. Moreover, apparently disagreement often triggered the emergence of new perspectives. That is, critical reasoning moves may be instrumental to creative reasoning, at least if discussants have an open attitude towards other perspectives.

Table 17

Causal hypotheses explaining the results of Wegerif et al. (1999), classified according to Mercer (2013)

Hypothesis	Causal mechanism
Appropriation / co-construction (weaker hypothesis)	Learning of dialogical group reasoning skills → improved group interaction → improved learning of specific problem-solving skills during group interaction → improved individual problem-solving performance
Transformation (stronger hypothesis)	Learning of dialogical group reasoning skills → improved individual general reasoning abilities (dialogical reasoning) → improved individual problem solving performance

Resnick et al. (2010) describe their vision of a *discursive classroom*. As discussed above, empirical results demonstrate that assigning a central role to discussion activities in the classroom promotes the learning of disciplinary knowledge in math and science. The more ambitious goal, however, is to achieve improvements on a wider scale, or, in the words of Resnick et al. (2010), "growing the mind[s]" of children. Humans have the unique capacity to employ language for their reasoning, so promoting certain forms of language use may have a direct bearing on reasoning skills more generally. Such far reaching changes can only be achieved through a socialization process in which the skills and habits of reasoned talk and thinking are regularly and broadly practiced. Hence, separate courses on critical thinking and logic may have only a limited impact. The better approach is to make discursive practice an integral part of school life across disciplines. This approach gives children the opportunity to experience and practice reasoning and discourse skills in a variety of genres which involve different facets and forms of thinking. Resnick et al. (2010) cite evidence demonstrating the impact discursive practice may have beyond the learning of specific knowledge and skills. In a study of another research team, British students participated in a two-year science education program in which discursive activities took a central role. Three years after the conclusion of the program, students not only showed significantly improved results in science but also in terms of their proficiency in English. The above mentioned Project Challenge, in which potentially talented children of low socioeconomic status took part in a four-year intervention program, also yielded some impressive results of transfer to other disciplines. A post-intervention test compared participants of the intervention program with other students of the same basic population (i.e., potentially talented students). The results showed significant and large effects of the intervention program not only for mathematics but also for general English skills.

2.8 Summary

Psychological research identified argumentation as a core component of informal reasoning. Informal reasoning comprises forms of reasoning employed when problems are ill-defined, open-ended, without one correct solution, and without a definite "recipe" for finding the best possible solution. Such problems typically have to cope with conflicting interests, values, beliefs, and objectives; they involve incomplete and uncertain information. Most real-world problems are of this kind. Argumentation is therefore of great practical relevance.

Generally, argumentative reasoning may be conceived of as the weighing of evidence and reasons in a framework of alternatives. Based on this conception, Kuhn (1991) identifies the following skills of argument: presenting one's own position, providing reasons and evidence, considering possible counter-positions, anticipating counterarguments, evaluating evidence, and rebutting counterarguments. Empirical studies of Kuhn (1991) and others disclose that people often exhibit critical weaknesses in these skills. People tend to selectively search and consider information confirming their own positions and beliefs. Similarly, they are often biased in their interpretations and judgments in a way that favors their own position. This phenomenon has been referred to as *confirmation bias* or *my-side bias*. Furthermore, people are often satisfied with the first explanation that makes superficially sense without contemplating alternatives or possible weaknesses of the chosen explanation. This phenomenon has been referred to as a *make sense epistemology*.

But what are the reasons why many people show such deficient forms of argumentation? Kuhn (1991) attributes the lack of argumentation skills to people's epistemic beliefs, that is, their theories about the nature of knowledge and knowing. Argumentation skills are most likely found in people holding an *evaluativist* stance, who view knowledge claims as products of the human mind that can be evaluated based on objective criteria and standards of reasoning. Epistemic beliefs may be seen in the context of general thinking dispositions. Thinking dispositions account for the fact that people may be generally able to engage in certain intellectual activities but decide not to do so. Non-evaluativists may simply not see much value in engaging in argumentative reasoning activities. Unavoidably, people who reject engaging in argumentative thinking miss opportunities to develop corresponding skills. So, low levels of epistemic belief and argumentation skills may go hand in hand. Empirical results provide evidence in line with this hypothesis. Besides argumentation skills and thinking dispositions, argumentative performance is assumed to critically depend on a sufficiently developed, domain-specific knowledge base. Finally, argumentation skills themselves may be subdivided in more general argumentative reasoning principles and domain-specific forms of argumentation. That is, engaging in argumentation in a particular field also requires knowledge about the specific procedures and standards of evaluation applied in that field, for instance, which kinds of evidence are seen as eligible.

Cross-sectional studies in developmental psychology suggest that basic argumentation skills are already present in relatively young children and further develop around the age of adolescence along different dimensions. While a certain level of proficiency may be reached quite naturally, e.g., through social interaction in the home environment and with peers, there is evidence suggesting that further improvements critically depend on formal education. For instance, Kuhn et al. (1988) found that college education—and not age—is the best predictor for argumentation performance from a certain age on. Moreover, researchers see a strong connection between argumentative reasoning and socially experienced language use. Educational institutions may then compensate for the lack of opportunities of underprivileged children to engage in conversational practice needed to develop higher-order reasoning and argumentation skills. However, argumentation skills still receive relatively little attention in schools and universities. Knowledge is often presented as a matter of fact rather than as the result of academic discourse and controversy. The skills of argument are rarely explicitly addressed outside of extracurricular debate clubs. A possible reason is a lack of knowledge, tools, and methods how to best teach argumentation and how to integrate new approaches with current practice.

Many argumentation-centered learning approaches are based on the social learning paradigm. Some of the reasons why the social learning paradigm became increasingly popular are: (1) Learning in social contexts is assumed to be particularly promising since the cognitive apparatus of humans has a special sensitivity towards social interactions. (2) Learning partners can serve as highly adaptive learning resources, models, or knowledge co-constructors. (3) The social context is the most realistic environment to practice social competencies, such as collaborative argumentation. Two social arrangements intensively discussed in the literature are learning from a more experienced person and learning in groups of peers. The benefits of the former approach are demonstrated, for instance, through the success of expert tutoring, which is known as one of the most effective forms of teaching. Key factors are the guidance and structuring (or scaffolding) provided by the tutor, and the learner's active uptake and use of the given support. Learning in groups of peers is more attractive from an economical perspective since one teacher can supervise one or multiple groups of learners at once; experienced learners may even regulate their group work without any external support. Moreover, collaboration among peers may have its own specific merits since learners can take a more responsible and active role in co-regulating their group work and learning. Yet, research shows that triggering productive group interactions requires a careful arrangement of the instructional setting. Early research tried to establish a direct link between aspects of the learning arrangements and learning success. For instance, it turned out that the "right" goal and incentive structure is important for successful learning in groups. The focus of interest shifted in later research in several respects: (1) Learning is typically attributed to the interactions between students rather than the individual dealing with task aspects. Therefore, later research increasingly focused on the role interactions take as mediators of learning. Moreover, the interactions between students open a direct window to observe the mechanisms of collaborative learning at work. (2) With the focus on interactions, research interests also shifted from cognitive to social processes, such as maintaining a joint understanding, negotiating the meaning of concepts, and co-constructing knowledge. (3) Not all tasks are equally suitable to trigger the rich sort of interactions mentioned above. For instance, some tasks can also be solved individually, through division of work, or with a sporadic exchange of information. Researchers therefore became increasingly interested in true group tasks involving ill-structured problems. Such tasks are particularly promising since their very nature requires that each group member gets actively involved. The complexity and openness of these tasks makes it necessary to explore, compare, and evaluate alternative options. The interactions between students involve a fair share of argumentive reasoning, which is seen as pivotal for learning.

Argumentation-centered learning arrangements contribute to three broad groups of learning objectives. First, the development of argumentation competency itself may be a learning goal. Studies show that the participation in argumentation activities has positive effects on argumentation skills. The activities may include group discussions about the question of what constitutes good and bad argumentation, or direct participation in argumentative discourse. Inexperienced learners require external support and guidance, which was provided in most of the discussed approaches through coaches or the teacher. Teachers / coaches may moderate and coordinate the discussion between learners, model desired discussion moves, and ensure that discussion norms are obeyed. Following the approach of scaffolding, teachers / coaches may gradually reduce their support and transfer responsibilities to learners. Substantial improvements of argumentation skills can only be realized through sustained engagement in corresponding learning activities over longer stretches of time. Second, the learning of subject matter content through argumentation may be a learning goal. Two types of interaction play a particularly important role. The first type is the joint elaboration of content, including building on what other participants said before and providing conceptual explanations. The second type comprises critical, dialectical moves, such as challenges, counter-arguments, and rebuttals. Research by Asterhan and Schwarz (2007, 2009) suggests that in particular critical moves have the potential to help learners identify and repair deeply rooted and persistent misconceptions. As indicated by several lines of research, conflicts are only conducive to learning when they are addressed in a nondogmatic and reasoned way with a focus on content rather than on ego. Empirical results indicate positive effects of argumentative activities in domains such as science and math. Third, engaging in argumentation may have positive effects on reasoning and problem solving more generally. This idea has its origins in the Vygotskian view that higher-order reasoning skills develop from participation in social interactions through a process of internalizing or appropriating reasoning patterns experienced in the social context. Thoroughly thinking about an issue can be conceived of as an internal argument between inner voices, an observation, which makes this view intuitively appealing. Several investigations suggest that sustained and guided participation in discursive activities indeed has positive effects on verbal and nonverbal skills not specifically addressed during the intervention.

Chapter 3
Computer-Supported Argumentation Learning

Over the last two decades, research has become increasingly interested in the use of computers for argumentation learning. For example, the above discussed intervention program used by Wegerif et al. (1999) also included computer-based activities. Computers can support argumentation-based learning in different ways. For instance, as mentioned above, a class of approaches to structuring the interactions of students called *scripted cooperation* (O'Donnell and Dansereau 1992) has become an important area of research. The use of computer software for scripting is particularly attractive since the structuring of student interactions can be realized or supported through specifically designed graphical user interfaces, computer-controlled pacing of activities, and system-generated instructions and prompts. Scripted cooperation (or *collaboration scripts*) constitutes one main group of approaches discussed in this chapter. Another group of approaches is based on the ability of computer tools to support, in a very easy and effective way, the individual or joint creation, modification, and scrutiny of structured representations of arguments. Yet another group of approaches discussed in this chapter relies on the ability of computer programs to automatically adapt their behavior to the learning process, imitating aspects of human tutoring or providing completely new and unique forms of interaction.

Section 3.1 discusses different research paradigms of educational technology. Two paradigms of particular relevance are *computer-supported collaborative learning (CSCL)* and *intelligent tutoring systems (ITS)*. Argumentative interactions have been identified as a key for learning in collaborative arrangements, which explains why argumentation emerged as a focal area of study in CSCL. The ITS community traditionally focused on analyzing, modeling, and supporting computer-based learning activities in relatively restricted domains. More recently, ITS researchers extended their scope to support collaboration and argumentation, which offers interesting new opportunities for enhancing CSCL system with adaptation mechanisms.

A recurring theme in the previous chapter was the importance of supporting students through structure and guidance. Section 3.2 and section 3.3 discuss two computer-based approaches to structuring and guiding student interactions, which became important topics in CSCL research. In particular, two theories will be addressed, which constitute the theoretical framework of the instructional approach designed as part of this doctoral thesis. Section 3.2 discusses the *theory of representational guidance*, which has its origins in research on graphical knowledge representations. This theory explains how the affordances of different representational notations affect the way learners jointly create, manipulate, and discuss external knowledge representations. By transforming the interactions between students in a specific manner, representational notations are assumed to crucially influence the achieved learning success. Section 3.3 discusses the *script theory of guidance in computer-supported collaborative learning*, which has its roots in research on collaboration scripts. Its basic assumption is that the way people collaborate is guided by their *internal scripts*, which are mental knowledge structures representing information about collaboration. An essential part of this theory is devoted to the question of how *external scripts*, that is, external representations of knowledge relevant to collaboration, can be utilized to guide collaborative activities and help learners develop more effective internal scripts and domain-specific knowledge.

Section 3.4 and section 3.5 discuss argumentation research originating from the fields of artificial intelligence and intelligent tutoring systems. The discussed research addresses the question of how argumentation learning systems can be enhanced with adaptive support functionality. Section 3.4 focuses on the automated analysis of argumentation, which is prerequisite for providing computational models that drive the adaptation of a system's behavior. Section 3.5 focuses on the adaptation strategies themselves. That is, once a system has the ability to analyze certain aspects of argumentation learning, it also needs some approach to adjust its behavior in a way that is conducive to the learning process. Section 3.6 summarizes the main insights of this chapter.

3.1 Educational Technology Approaches

The use of technology for educational purposes has a long tradition, including approaches to integrate film, radio, and television in the classroom (Cuban 1986, cited in Koschmann 1996). It was the advent of computer technology that intensified research interest considerably and gave rise to specific research paradigms of educational technology. Koschmann (1996) discusses four main paradigms, each entailing specific assumptions about learning and teaching, pursuing specific instructional and research objectives, and employing specific technological and research approaches. In particular, these are: computer-assisted instruction (CAI), intelligent tutoring systems (ITS), Logo-as-Latin, and computer-supported collaborative learning (CSCL).

- *Computer-assisted instruction (CAI)* emerged in the 1960s. Applications in this tradition are restricted to presenting teaching materials in a logical sequence to learners, according to instructional goals and didactical considerations (e.g., some piece of content requires another piece as a prerequisite). Practical exercises are realized through programmed *drill-and-practice* (Stahl et al. 2006), that is, the computer poses a question, the learner inputs an answer (e.g., multiple choice or fill-in-the-blank), and, depending on the correctness of the answer, a more challenging question is presented, and so forth. Developers of CAI programs are often educational practitioners enabled to create contents through courseware authoring programs. Therefore, corresponding CAI software has a strong practical orientation and reflects prevailing traditional views on instruction and learning. In particular, learning is viewed as the (passive) acquisition of knowledge facts and instruction as the transmission of those facts. CAI can be associated with a behavioristic perspective, focusing on stimulus-response relationships: Presenting teaching materials or feedback (the stimulus) leads to improved performance (the response). Internal mental processes are typically not considered. Corresponding research is mainly interested in *instructional efficacy*, that is, whether knowledge gains superior to some control can be realized through some sort of CAI.
- *Intelligent tutoring systems (ITS)* emerged in the 1970s. ITS research has its origins in artificial intelligence and cognitive science. Its vision and guiding theme is to emulate human tutors by posing challenging, multi-step problems and providing feedback and hints based on the learner's problem-solving actions, knowledge level, and misconceptions. ITS research puts particular emphasis on the learner's mental representations of knowledge and problem-solving (*cognitivism*). Mental structures and processes are modeled using computational *student models*. For instance, *model tracing* (Corbett and Anderson 1995) is a computational method to model mental operations during problem solving. The problem-solving process is often conceived of as the traversal through a *problem space*, which includes an initial state, solutions states, and a limited set of operations to traverse between states. The system

analyzes each problem-solving step (i.e., each user input) to decide whether the step is *correct* or *incorrect*, based on a set of rules representing the system's ideal model of problem solving. More sophisticated approaches utilize *buggy rules* to identify *misconceptions* or *mind bugs* learners may have. If a step is incorrect, the system provides feedback and hints to put the learner on the right track again. Model tracing is often used in concert with another technique called *knowledge tracing* (Corbett and Anderson 1995). Knowledge tracing is an approach to modeling the probabilities of specific *skills* (or *knowledge components*) being mastered by a learner, based on the analysis of his problem-solving steps. The ITS approach has notable advantage over the CAI approach. First, the ITS-modeling machinery allows the system to make more informed decisions and give support tailored to the specific needs of learners. For instance, ITS systems may select problems based on the learner's mastery profile and provide feedback tailored to the learner's misconceptions. Some ITS systems employ *domain reasoners* to determine, on the fly, whether arbitrary inputs are correct or not. This is a critical advantage over CAI systems, which are typically far less flexible and restricted to checking for predefined answers in a database. Second, ITS systems trace problem-solving on a more fine-grained level and provide feedback for individual problem steps (VanLehn 2006). VanLehn (2011) presents a meta-analysis comparing different kinds of instructions, grouped according to the level of granularity of the interaction involved. He comes to the result that interactions at the step level (the interaction granularity of most ITS systems) are superior to interactions at the answer level (the interaction granularity of most CAI systems) in terms of the achieved learning gains. As a possible explanation, VanLehn (2011) proposes that the interaction granularity in answer-level systems is too coarse. The step from the question to the final answer is simply too big, involves too much reasoning, so students often resort to guessing or quitting. Conversely, ITS systems provide tailored feedback and hints on intermediary steps, which enables learners to accomplish the step themselves while extending or self-repairing their knowledge bases. However, building ITS systems is a complex matter due to the intricacies of student modeling, which requires representing and updating computational representations of mental structures and processes not immediately accessible to researchers and the system. Therefore, most ITS research has focused on relatively narrow, easy-to-formalize, and procedural domains such as arithmetic and calculus. While earlier ITS research was strongly focused on *instructional competence*, that is, how good computer models can emulate aspects of a real tutor (e.g., diagnosing misconceptions and assessing knowledge), the question of instructional efficacy gained considerable importance more recently.

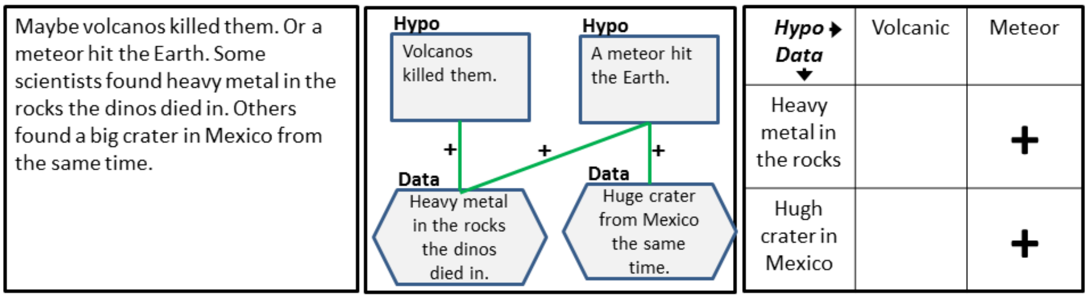
- *Logo-as-Latin* describes a class of approaches that emerged in the 1980s, inspired by constructivist learning theories. ITS systems may be designed with the constructivist ideal in mind too, e.g., the system may prompt learners to reflect on their errors or to self-explain a solution step—as noted by VanLehn (2011), system-generated scaffolding in ITS systems may help students to do most of the reasoning themselves. However, the flow of interactions is still largely controlled by the system along predefined pathways. ITSs typically classify student actions as right or wrong, based on the body of domain knowledge encoded in the system, then try to remediate behaviors not conformant with their model, an approach criticized as the “arrogant ‘tutor knows best’ style of ITSs” (Self 1990). Approaches under the Logo-as-Latin paradigm give learners much more freedom and control over their activities and learning. Corresponding instructional technologies (e.g., micro-worlds and simulations) take a passive role in providing a playground or environment for learners to freely explore concepts of interest, carry out experiments, and test out ideas. Many approaches are based on the *Logo* programming language, which enables young children to experiment with programming concepts, such as loops and variables. The main goal, however, is not to promote programming skills but rather to foster self-regulated learning and problem-solving skills more generally. To emphasize these more general objectives of the approach, Koschmann (1996) proposes the term *Logo-as-Latin* in analogy to Latin, whose learning was in former times assumed to improve general intellectual abilities. Since the main goal is to support general skills, evaluation studies often focus on questions of *transfer*.
- *Computer-Supported Collaborative Learning (CSCL)* was established in the 1990s as an independent field of research. Two developments contributed to the emergence of CSCL (Stahl et al. 2006). First, rapid technological advances and the widespread adoption of personal computers and the Internet raised the question of how these new technologies could be employed to improve education and prepare children for the digital age. Second, as discussed, the learning sciences recognized the great potential of collaborative learning approaches to promote deep content learning and general collaboration and thinking skills. In contrast to CAI and ITS research, CSCL sees the interactions between peers—not the instructions and support provided by the system—as the primary source of learning (Stahl et al. 2006). Yet, the role of technology is

> typically not reduced to a pure medium of discourse. Rather, a main portion of CSCL research investigates ways how technology can facilitate, guide, and scaffold high-quality interactions between learners. Two prominent ideas how such guidance can be realized—specific knowledge representation formats and collaboration scripts—will be discussed next. CSCL essentially expands on the theories of learning and instruction adopted by collaborative learning researchers, as described in section 2.6. CSCL research puts particular emphasis on peer interactions, group-level processes, such as knowledge co-construction, and true group tasks involving ill-structured problems. Therefore it is not surprising that argumentation is one of the "flash themes" in CSCL (Stahl 2007). CSCL research employs and partly mixes methodologies from different traditions, including experimental, descriptive, and iterative design approaches (Stahl et al. 2006).

It can be observed that, over the years, the different fields extended their scope and imported questions, ideas, and methods from one another. For instance, a number of more recent ITS approaches do not try to emulate an expert tutor anymore but simulate, for instance, learning companions (Goodman et al. 1998) and tutees to be taught by a learner (Walker et al. 2011). ITS systems nowadays do not solely focus on specific, narrowly focused skills but also try to promote self-regulation and metacognition (Azevedo et al. 2010), help-seeking behaviors (Aleven et al. 2006), and collaboration (McManus and Aiken 1995). ITS researchers transcended the boundaries of formal knowledge domains to target more ill-defined and open ones, which are notoriously hard to tackle using traditional ITS methods and approaches (Lynch et al. 2009). Researchers of the CSCL community increasingly realize the potentials of adaptation technologies—the classical province of ITS research—to support collaborative learning processes (Fischer et al. 2013). There is new research in the cross-section between ITS and Logo-as-Latin. For instance, the MiGen project uses ITS methods to (unobtrusively) support exploratory learning (Noss et al. 2012). The Metafora project takes the idea even one step further by researching ways to automatically support collaborative exploratory learning, combining ideas and techniques from the ITS, Logo-as-Latin, and CSCL traditions (Dragon et al. 2013).

3.2 Representational Guidance

Knowledge can be represented in different ways. A classical way of representing knowledge is the use of plain text. For instance, an essay may explain a specific matter of fact or put forward an argument to justify some opinion. Other representational formats may be employed to make certain information more accessible to the reader. For instance, text books often include diagrams and charts to illustrate certain aspects of interest visually. Knowledge representations may not only be used to present information in a static fashion. Rather, users, or learners, may actively create and manipulate knowledge representations. For instance, a vivid field of educational research is the use of concept mapping, a method to graphically represent the relationship between concepts in a domain of instruction (Novak 1990). Computer-based representational tools can support the working and learning with knowledge representations by enabling, or facilitating, many relevant tasks, such as exploring, creating, modifying, storing, organizing, analyzing, and sharing of knowledge representations.



Maybe volcanos killed them. Or a meteor hit the Earth. Some scientists found heavy metal in the rocks the dinos died in. Others found a big crater in Mexico from the same time.

Hypo
Volcanos killed them.
Hypo
A meteor hit the Earth.
+
+
+
Data
Heavy metal in the rocks the dinos died in.
Data
Huge crater from Mexico the same time.

Hypo ▸ *Data* ▾	Volcanic	Meteor
Heavy metal in the rocks		+
Hugh crater in Mexico		+

Figure 5: Knowledge representations based on different representational notations: text (left), graph (middle), table (right). Adapted from Suthers (2003).

A considerable body of research has focused on the question whether specific *representational notations* (i.e., systems of graphical or linguistic symbols used to convey meaning) can promote productivity, insight, and learning. Figure 5 shows three exemplary knowledge representations employing different notations—a text, a graph, and a table based notation—which are used to represent essentially the same information:

- T*ext-based notations* (Figure 5, left) represent information by sequentially arranging words into meaningful sentences, according to grammatical rules. Natural language text is probably the most common notational format for representing knowledge in a persistent way.

- *Graph-based notations* (Figure 5, middle) represent information by decomposing a body of knowledge in isolated knowledge chunks (represented by nodes or boxes). Knowledge chunks are classified according to semantic categories (represented by graphical shapes [e.g., rectangle, hexagon] and / or semantic labels [e.g., *hypo* or *data*]). Relations between knowledge chunks are represented by links or arcs, which again, are classified according to semantic categories (e.g., a plus sign [+] may signify a consistency relations, a minus sign [-] may signify an inconsistency relations).

- *Table-based notations* (Figure 5, right) represent information in a two-dimensional matrix. Again, knowledge is decomposed into isolated knowledge chunks, which are classified into one of two semantic categories (in the example: *hypo* and *data*). Elements of one category constitute the x-axis (e.g., hypotheses). Elements of the other category constitute the y-axis (e.g., data). Each table cell classifies the relation between the corresponding hypothesis and data element according to semantic categories, where applicable (e.g., a plus sign [+] may signify a consistency relations, a minus sign [-] may signify an inconsistency relations).

The effects of different representational notations may be explained in terms of the constraints and saliences they possess (Suthers 2003). *Constraints* are limits on what can principally be expressed within a representational notation. Essentially, constraints delimit the expressive power of a notation. For instance, a text can represent many different relations, which are available in the lexicon of the used natural language, while the graph and table notations discussed above only distinguish between consistency and inconsistency relations. *Saliences* direct the attention to certain aspects of a representation, possibly at the expense of other aspects. Thus, saliences can help users capture specific aspects of encoded meaning quicker, facilitate the search of information, or invite users to perform specific actions, based on *perceived affordances* (Norman 1988). For instance, compared to text, graphs and tables make it easier to identify individual knowledge chunks and their interrelations. An empty table cell *affords* users to think about possible relations between corresponding knowledge chunks. Saliences not only depend on the notation itself but also on the perceptual architecture of the agent dealing with the representation. Essentially, saliences arise from automated, rather than controlled, perceptual processes of the agent.

An important aspect with respect to representational guidance is the specific vocabulary used within a representational notation, for instance, the labels of nodes (e.g., *hypothesis*, *data*) and links (e.g., *supports*, *opposes*). Such *ontologies* define the conceptual space in which users operate. While actively engaging with important concepts reflected in the ontology of a notation, learners may gradually learn to think in terms of these concepts. For instance, in order to classify knowledge chunks as data and hypotheses, students have to understand the meaning of and difference between these two important notions, a crucial prerequisite for scientific reasoning.

In his *theory of representational guidance*, Suthers (2003) focuses specifically on the influence different representational notations may have on collaborative learning processes. The basic scenario considered in Suthers' research is that multiple learners jointly construct a shared knowledge representation using a specific representational notation. He identifies three roles representations may play specific to collaborative settings. First, representations may initiate negotiations of meaning. Since the representation is shared between multiple learners, changes must be coordinated and agreed. For instance, adding a new data element to a graph may spark a discussion about whether the represented information is actually *data* or may rather be classified as a *hypothesis*. By negotiating the meaning of the two categories *data* and *hypothesis*, learners may develop a better understanding of the difference between both categories. Second, created representations can serve as target of pointing acts. That is, learners can more easily refer to ideas previously dealt with by pointing at elements within the created representation. The act of pointing can take different forms, for instance, physical pointing (in collocated settings), verbal references (e.g., each box in a diagram may have a referable sequence number), or the use of specific awareness tools (e.g., a cursor icon that can be placed on elements of the representation). Thus, processes of communication, negotiation of meaning, and knowledge elaboration may be facilitated; the quality of collaboration may increase. Finally, shared representations may be considered as group memories of previous ideas and agendas for future work (e.g., through the absence of elements in the shared representation, e.g., an empty table cell). Again, processes of knowledge co-elaboration may benefit, since participants are reminded of prior ideas, which could be reconsidered in the current context and put into relation with more recent ideas.

Empirical evidence is provided by Suthers and Hundhausen (2003), who compared different representational notations (in particular: *text*, *graph*, and *table*) in terms of their influence on collaborative learning processes. Some predictions derived from an analysis of the constraints and saliences of the different notations could be confirmed. In particular, the more structured and constrained representational notations *graph* and *table* led to significantly more knowledge co-elaboration compared to *text*. They also found that affordances of representational notations can lead to problems. For instance, learners in the *table* condition considered many spurious relations between data elements and hypotheses. One explanation is a strong prompting effect of empty table cells, which potentially tempted learners to also include questionable relations.

Considered from a system design perspective, the task of the designer is to exploit representational biases inherent to specific notations to create a condition of *representational guidance* (Suthers 2008), which promotes, in a targeted way, social and / or cognitive processes conducive to learning. Figure 6 sketches the idea of designing for representational guidance. Designers develop specific *representational notations* and make these notations available through some *representational tool* (e.g., through a palette from which users choose available box and link types). Learners then use this representational tool, individually or jointly, to read, create, and manipulate *representational artifacts*, that is, specific products of representational activities. While doing so, their thinking and acting is guided by the representational notation built into the representational tool, based on the saliences and constraints of the employed notation.

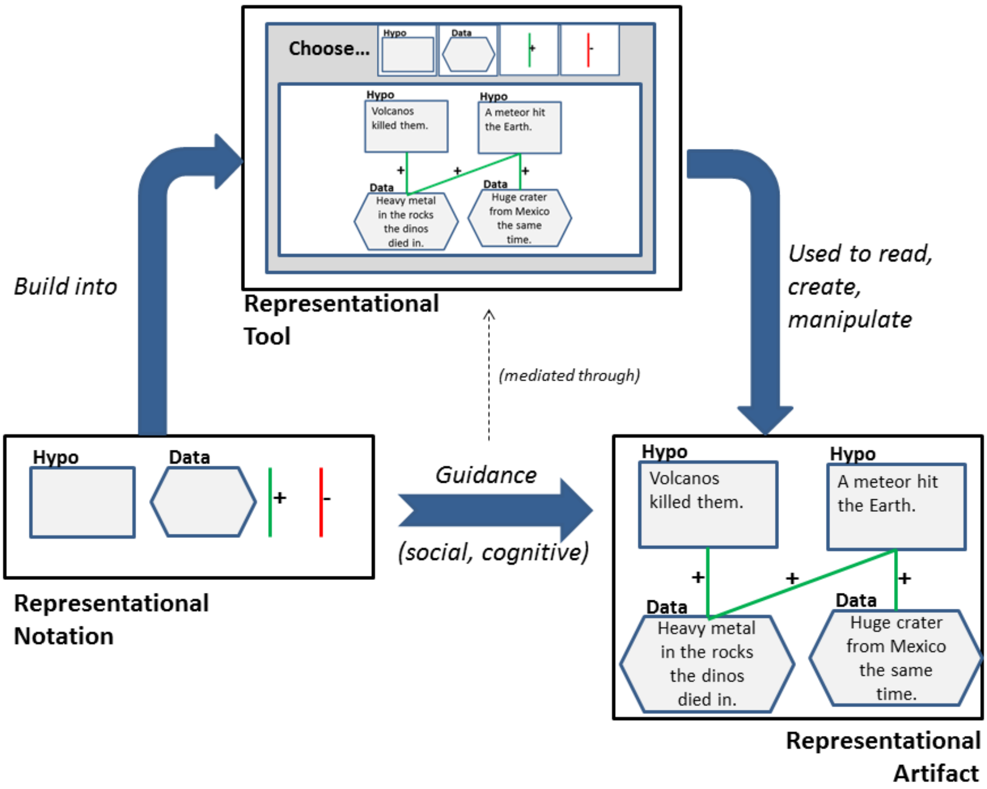
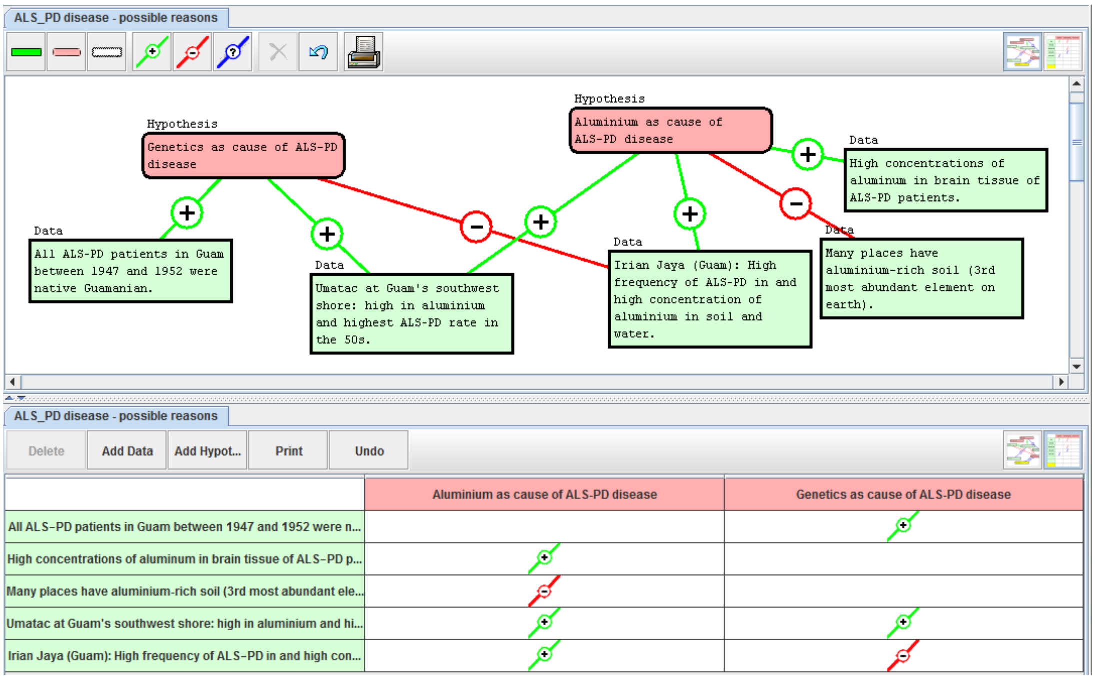

Choose...
Hypo
Data
Volcanos killed them.
A meteor hit the Earth.
Heavy metal in the rocks the dinos died in.
Huge crater from Mexico the same time.
+
-
Build into
Representational Tool
Used to read, create, manipulate
(mediated through)
Guidance
(social, cognitive)
Representational Notation
Representational Artifact

Figure 6: Representational guidance. Adapted from Suthers (2003).

3.2.1 Representational argumentation systems

Probably the most common representational approach to argumentation learning is *argument diagramming* (also called *argument mapping*). Argument diagrams typically employ the graph-based approach discussed above. That is, the structure of arguments is represented through node-and-link graphs. Figure 7, upper panel, shows an *evidence map* within the Belvedere system (Suthers et al. 2001). Evidence maps focus on one of the most central notions in scientific argument: the distinction between hypotheses and evidence. Propositions are represented as nodes (*hypothesis* and *data* elements), and relations between propositions are represented as graphical links (e.g., a piece of data supports [+] or opposes [-] a hypothesis). In this specific instance, the evidence map is used to analyze possible reasons for the ALS-PD disease, which occurred at an unusually high rate on the island of Guam. The two hypotheses explored in the map are that the disease is caused by aluminum or by a genetic disposition.



Hypothesis
Genetics as cause of ALS-PD disease

Hypothesis
Aluminium as cause of ALS-PD disease

Data
All ALS-PD patients in Guam between 1947 and 1952 were native Guamanian.

Data
Umatac at Guam's southwest shore: high in aluminium and highest ALS-PD rate in the 50s.

Data
Irian Jaya (Guam): High frequency of ALS-PD in and high concentration of aluminium in soil and water.

Data
High concentrations of aluminum in brain tissue of ALS-PD patients.

Data
Many places have aluminium-rich soil (3rd most abundant element on earth).

	Aluminium as cause of ALS-PD disease	Genetics as cause of ALS-PD disease
All ALS-PD patients in Guam between 1947 and 1952 were n...		+
High concentrations of aluminum in brain tissue of ALS-PD p...	+	
Many places have aluminium-rich soil (3rd most abundant ele...	−	
Umatac at Guam's southwest shore: high in aluminium and hi...	+	+
Irian Jaya (Guam): High frequency of ALS-PD in and high con...	+	−

Figure 7: Hypotheses-evidence relations represented within the Belvedere system (Suthers et al. 2001) as node-and-link graph and table. Data represented in figure from Suthers and Hundhausen (2003).

Evidence maps are just one example of node-and-link notations employed in argumentation systems. Different sets of node and link labels may be used to represent other categorical systems (or *ontologies*). Ontologies may be domain-independent, e.g., an ontology with *statement* boxes and *support* and *oppose* links; a widely used relatively generic ontology is the Toulmin (1958) model. Other ontologies incorporate domain-specific aspects. For instance, to represent legal arguments, models developed by Wigmore (1931; used in Araucaria [Reed et al. 2007]) and Ashley (1990; used in LARGO [Pinkwart et al. 2009]) have been used. Other ontologies again have been designed to represent arguments about planning and design problems (Rittel and Webber 1973; used in gIBIS [Conklin and Begeman 1988]). Ontologies might be large and allow fine-grained distinctions, or might be restricted in size and coarse-grained in meaning, making it easier and less error-prone to use. For instance, Suthers et al. (2001) considerably simplified the ontology of Belvedere after realizing that students had problems using the more expressive ontology. In conclusion, the design of an appropriate ontology is highly critical. It requires a careful analysis of student prior knowledge and experience, learning goals, and the concrete application context in order to find an optimal trade-off between expressiveness, learnability, and usability. Because there is no one-size-fits-all solution, systems should allow an easy and flexible configuration of ontologies.

While argument diagramming has a long tradition (e.g., Wigmore 1931), computer-based argumentation systems have become increasingly popular since the mid-1990s, resulting in systems such as the aforementioned Belvedere (Suthers et al. 2001), Reason!Able (van Gelder 2002), and Digalo (Schwarz and Glassner 2007). Such tools have been used in a variety of ways and in different domains, for instance, to analyze existing legal arguments (Pinkwart et al. 2009), to outline arguments in preparation for essay writing (Janssen et al. 2010), or to discuss given contentious questions (Schwarz and Glassner 2007).

The main feature of argument graphs is that argument structures are represented visually and explicitly. In contrast to less explicit formats, such as prose, graphs allow students to immediately see how lines of reasoning evolve, step by step, without having to infer argumentative relations (van Gelder 2005). By way of their explicitness, graphs can help students see faulty reasoning. While prose and chat are linearly arranged, argument graphs allow multilevel hierarchies (or networks), thus better match the hierarchical structure of many arguments (van Gelder 2005). Cognitive processing can be further facilitated through graphical elements such as colors, lines, and shapes (van Gelder 2002). While expressing knowledge in a highly structured format unavoidably involves cognitive overhead, e.g., through a "premature commitment to structure," it has the potential to trigger processes of reflection and deeper understanding (Buckingham Shum et al. 1997). In particular, the specific category systems used in argument diagramming tools can focus students' attention on important concepts of argumentation and encourage reflection about these concepts, e.g., the distinction between hypotheses and data in scientific arguments (Suthers and Hundhausen 2003).

On the downside, node-and-link graphs may become unwieldy, especially in synchronous collaborative settings when many contributions are created in rapid succession. Sometimes the result is a "spaghetti" image, which is hard to read and follow (Loui et al. 1997). In general, the quality and readability of argument diagrams depends on how skillfully users organize and spatially arrange information.

Graphs arranged according to Gestalt principles, such as symmetry, continuation and proximity, have been shown to be more useful learning resources, leading to higher learning gains (Dansereau 2005). Finally, while the typical arrangement of graphs according to logical and thematic relationships helps students to focus on the underlying argument structure, the temporal sequence of contributions is often not obvious, making it hard to identify recent contributions. Notably, many of these issues can be remediated or softened through system functions, for instance, orientation support (e.g., mini-maps, search functions) and awareness support (e.g., displaying creation timestamps, highlighting recent contributions).

Empirical studies show that learning with argument graphs can have positive effects on the quality of learning processes (e.g., increased elaboration [Suthers and Hundhausen 2003], improved quality of causal reasoning [Easterday et al. 2009]) and learning outcomes (e.g., critical thinking and argumentation skills [Harrell 2008; Twardy 2004], causal reasoning skills [Easterday et al. 2009]). Positive effects are reported in a variety of domains, including philosophy (Harrell 2008; Twardy 2004), the Law (Pinkwart et al. 2009), policy deliberation (Easterday et al. 2009), science (Suthers and Hundhausen 2003), and history (Janssen et al. 2010). Positive results have been achieved both in collaborative settings (Janssen et al. 2010; Suthers and Hundhausen 2003) as well as in individual settings (Easterday et al. 2009; Pinkwart et al. 2009).

An alternative form of argument diagramming is to employ a container-based notation. Figure 8 shows such a container representation of an argument within the SenseMaker system (Bell 1997). Claims are represented as visual frames. Nested frames represent supporting arguments. Nested hyperlinks represent supporting evidence, which can be accessed by clicking on the hyperlink. The colored dots indicate how strong students rate the different pieces of evidence. Students use the system for scientific inquiry to investigate alternative hypothesis regarding given questions (e.g., "How Far Does Light Go?") based on available online resources. Students may use SenseMaker individually or in small groups in front of the same computer.

Other approaches use simple lists of pro and con arguments, or tables (or matrices) to represent argumentative relations between propositions. Figure 7, p. 103, lower part, shows such a tabular representation in Belvedere (Suthers et al. 2001). Each column represents a specific hypothesis. Each row represents a specific piece of data. Cells indicate the evidential relation between the respective (hypothesis / data) pair, showing whether the data supports or opposes the hypothesis. Table and graph in Figure 7, p. 103, represent exactly the same argument. Tables are generally less expressive than graphs. For instance, argumentation sequences with three or more elements cannot be directly expressed in a table. Yet, on the positive side, the tabular format appears to be particularly useful to systematically (and exhaustively) check pairwise relations, e.g., which piece of evidence supports or opposes which hypothesis.

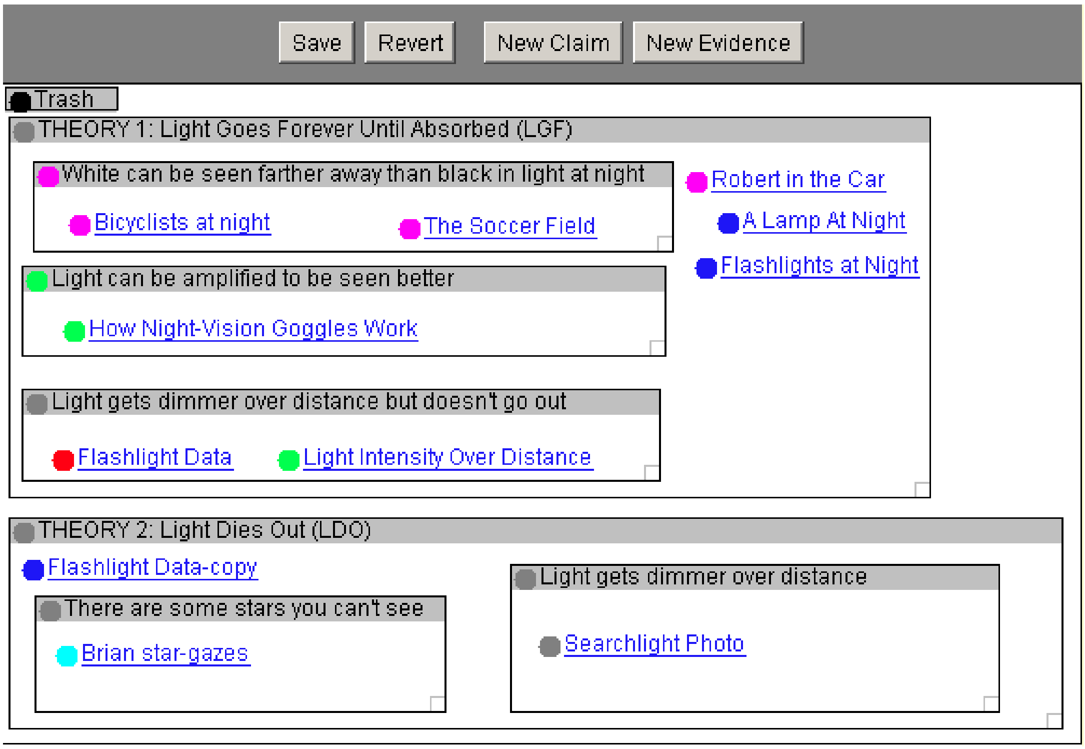

Save
Revert
New Claim
New Evidence
Trash
THEORY 1: Light Goes Forever Until Absorbed (LGF)
White can be seen farther away than black in light at night
Bicyclists at night
The Soccer Field
Robert in the Car
A Lamp At Night
Flashlights at Night
Light can be amplified to be seen better
How Night-Vision Goggles Work
Light gets dimmer over distance but doesn't go out
Flashlight Data
Light Intensity Over Distance
THEORY 2: Light Dies Out (LDO)
Flashlight Data-copy
There are some stars you can't see
Brian star-gazes
Light gets dimmer over distance
Searchlight Photo

Figure 8: Argument represented within the SenseMaker system (Bell 1997) in a container format. From Scheuer et al. (2010).

In summary, representational approaches may be used both in individual and in collaborative learning contexts, that is, students create and discuss argument representations in pairs or small groups. Different notations, such as node-and-link graphs, containers, tables, and lists, can be used to represent structural and semantic aspects of arguments. The question arises how the different notations impact and influence learning processes and learning outcomes. Important insights can be gained from Suthers' (2003) theory of representational guidance, which is based on the observation that notations differ in terms of salience of knowledge units and constraints on expressiveness. For instance, graphs and tables can represent evidential relations (as graphical links and table cells, respectively) in a more explicit way compared to written text. Therefore, evidential relations are more salient in graphs and tables, and consequently, students are more likely to elaborate on such relations when they create and discuss arguments. So to speak, the representation draws the attention of students to specific aspects and thus biases their activities and discussions in specific ways. The task of the system designer is hence to turn representational biases into representational guidance, that is, to create user interfaces that purposefully steer collaboration in "fruitful" directions (Suthers 2008).

3.3 Scripted Collaboration

High-quality interactions between inexperienced learners typically do not occur naturally, without external support. As discussed, early research in cooperative learning identified the goal and incentive structure as a possible means to motivate group members to interact in supportive ways with one another. Yet, even if people are highly motivated, they may still lack essential skills in productively collaborating and communicating with one other. Later research experimented with approaches to influence the interactions between learners in a more direct and targeted way. Such approaches—typically referred to as *collaboration scripts*—can be conceived of as scaffolds that operate on the interaction / process level rather than on the content / conceptual level (Kollar et al. 2006). Under the label of *scripted cooperation*, O'Donnell and Dansereau (1992) investigated collaboration scripts in face-to-face settings. The advent of networked computer technology then sparked interest in exploiting the high potential of computer-based collaboration scripts to structure and guide interactions within learning groups (overviews are provided, for instance, in Kobbe et al. 2007; Kollar et al. 2006; Weinberger 2011).

Kollar et al. (2006) provide an overview of face-to-face and computer-based collaboration scripts. An example for a face-to-face approach they discuss is the MURDER script by O'Donnell, Dansereau, and colleagues (MURDER is an acronym denoting a set of activities involved in the script). The script aims at helping pairs of learners to better understand a given text through joint activities. Learners go through the text passage by passage. First, each learner reads the passage individually. Then, one learner takes the role of a "recaller" who has to summarize the passage as complete as possible. The other learner takes the role of a "listener" who tries to identify and fix misconceptions and omissions. Finally, both partners jointly elaborate on the passage, e.g., by making connections to their prior knowledge. The roles of recaller and listener are switched for the next passage. An example for a computer-based approach is the learning protocol approach by Pfister and Mühlpfordt (2002). Groups from three to five learners and a tutor use a computer-based communication interface to discuss topics in geology and philosophy. The communication interface provides several means to structuring the dialogue. First, learners must explicitly indicate which previous message (or message part) they are referring to by means of graphical arrows. Second, learners must choose between three different message types to indicate the intention of their messages (question, explanation, or comment). Third, the system provides *floor control* functionality by automatically regulating the sequence of turn-taking between discussants (e.g., after a question the system gives the floor to tutor to respond).

Kollar et al. (2006) propose a five-component scheme to describe collaboration scripts:

1. *Objective*: Collaboration scripts are instructional scaffolds designed with specific instructional goals in mind. The MURDER script pursues the goal to support learners in (a) improving their text understanding and (b) developing strategies for learning from texts, such as generating explanations. The learning protocol aims at helping learners improve (a) their coordination behavior during the discussion and (b) their conceptual knowledge about target concepts in geology and philosophy.
2. *Activities*: Collaboration scripts specify specific activities conducive to achieving given learning goals. One of the activities in the MURDER script is to explain the content of text passages to the learning partner. The learning protocol prompts learners to use specific discussion moves (questions, explanations, comments) and to coordinate their interactions by explicitly indicating which previous message they are replying to.
3. *Sequencing*: Collaboration scripts may specify the chronological order of activities. The MURDER script essentially defines a three-step sequence comprised of reading, summarization, and elaboration. The learning protocol sequences the order in which learners contribute to the discussion (without requiring a specific sequence of message types).
4. *Roles:* Collaboration scripts may assign specific roles to learners. In the second step of the MURDER script, one learner is assigned the role of a recaller and the partner is assigned the role of a listener. Roles are alternated for each new passage. The learning protocol does not make use of explicit roles.
5. *Type of representation:* Collaboration scripts use different means to present instructions to learners or to impose structure on the learners' interactions.

> One variation of the MURDER script provided instructions on paper to learners. Another variation was that learners were trained in the procedures of the script before actually engaging in collaboration. That is, they employed internalized mental versions of the script during the learning activity. The learning protocol implements script elements in the user interface by displaying information and restricting interactions (who is next to contribute, labeled message types, graphical arrows to indicate messages referred to).

Dillenbourg (2002) describe different *coercion degrees* of collaboration scripts, ranging from induced scripts (the script implicitly conveys expectations regarding desired patterns of interaction), to instructed scripts (expectations are made explicit through instructions before collaboration takes places), to trained scripts (students are trained in target behaviors before collaboration takes places), prompted scripts (students are prompted to follow targeted behaviors during collaboration), and follow-me scripts (the learning environment enforces specific forms of interaction, e.g., in the learning protocol approach discussed above, only selected students can technically enter messages in the chat box to enforce a specific order of participants). According to Dillenbourg (2002), the "right" coercion degree poses a design dilemma for collaboration scripts. On the one hand, low coercion degrees may be ineffective since students may decide to not following the script. On the other hand, high coercion degrees may restrict students too much, run counter to the very idea of collaborative learning, and decrease students' motivation.

Fischer et al. (2013) observe that, while a number of promising empirical results regarding collaboration scripts exist, a theoretical basis is missing to systematically explain and predict effects across different scripting approaches. To fill this gap, they propose a scripting theory of guidance based on the body of available evidence. The basic assumption is that collaborative behavior is guided by mental knowledge structures, called *internal collaboration scripts*, which are similar in structure to *external collaboration scripts*. Following a theater metaphor, Fischer et al. (2013) describe the following layers of collaboration knowledge (i.e., internal script components):

1. *Play* is the topmost element and includes the overall goal of a collaborative activity (e.g., conducting a *critical discussion* aimed at jointly finding the most reasonable solution). It involves knowledge about typical *scenes* of the play, the sequence of these scenes, and the *roles* participants assume.

2. *Scenes* describe specific situations that can occur during a play (e.g., a critical discussion unfolds into the scenes *confrontation*, *opening*, *argumentation*, and *conclusion*). Scenes themselves may unfold into multiple sub-scenes.
3. *Roles* are subcomponents of a play and typically extend across several scenes (e.g., a critical discussion involves the roles of a proponent and an opponent of a given claim each trying to convince the other from one's own position).
4. *Scriptlets* describe specific activities within a scene and their meaningful sequencing (e.g., making a reasoned argument during an *argumentation* scene may include scriptlets to present a claim, to provide grounds, and to cite evidence). The specific scriptlets employed as part of a scene also depend on the roles learners assume.

Collaboration scripts take a twofold role by guiding both the learner's understanding of and acting in collaboration practices [*internal script guidance principle*]. Learners dynamically compose collaboration scripts from *play*, *scene*, *role*, and *scriptlet* components available in their internal repertoires [*script configuration principle*]. The way how the script is ultimately composed depends on the learner's goals and situational constraints and affordances (e.g., representational constraints and saliences as discussed above). The dynamic configuration of components allows learners to flexibly respond to a variety of situations, including situations they are initially unfamiliar with. Learners consolidate their collaboration knowledge and develop higher-level components (e.g., a specific play component) through repeated application of an effective configuration of lower-level components (e.g., a sequence of scene components the play component is comprised of) [*script induction principle*]. If the configuration does not lead to the expected success, learners are likely to modify the configuration of the internal script [*script reconfiguration principle*]. The transactive application of knowledge within a scripted or unscripted CSCL practice determines the extent to which this practice is conducive to domain knowledge learning [*transactivity principle*]. That is, CSCL practices involving a fair amount of transactive moves (i.e., reasoning on the reasoning of others) are expected to lead to superior domain knowledge gains.

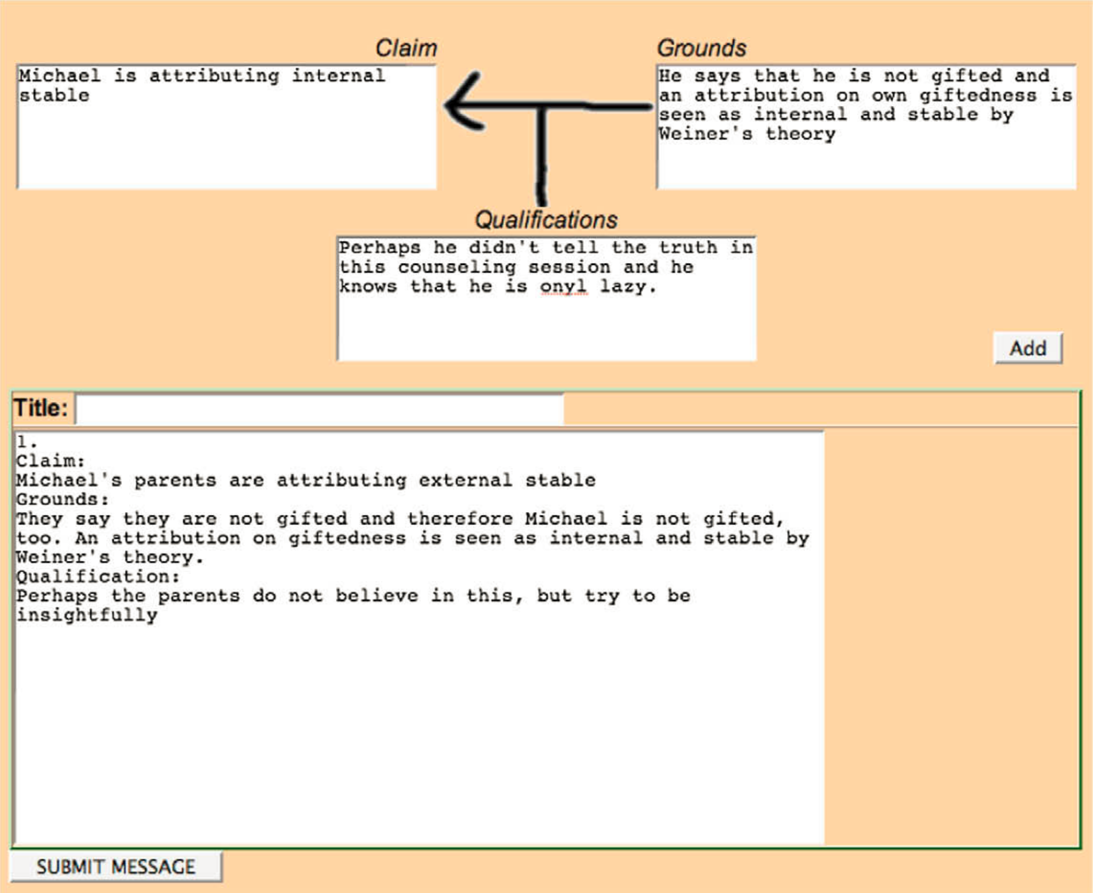
External collaboration scripts have the potential to empower learners to engage in collaboration practices that would otherwise be beyond reach [*external script guidance principle*]. On the one hand, they may inhibit the application of ineffective internal scripts a learner conventionally uses. On the other hand, they may help learners organize known lower-level script components into effective higher-level components the learner does not routinely use yet. External scripts may operate at different levels to scaffold collaboration. *Play scaffolds* explicate the overall goals of a collaborative practice. *Scene scaffolds* explicate which scenes are involved in a play and in which order. *Role scaffolds* explicate knowledge about specific roles important in a play and assign these roles to learners. Finally, *scriptlet scaffolds* explicate which scriptlets are relevant in a scene and in which order to apply these scriptlets. External collaboration scripts are most effective (in terms of both, collaboration practices and domain knowledge) when they build upon available internal script components at the highest level possible [*optimal script level principle*].

The script theory of guidance builds upon well-accepted notions in cognitive science [the schema-based theory of dynamic memory proposed by Schank (1999)] and education [scaffolding and learning in the zone of proximal development (Vygotsky 1978)]. It accounts for a large body of findings in collaboration script research. It offers a unified conceptual framework that provides a direct mapping between mental structures (internal scripts) and instructional structures (external scripts), and formulates principles how the latter influence the former. Yet, the proposed theory is just a first step *towards* a more complete and detailed account of scripted collaboration. As discussed by Fischer et al. (2013), important research topics are adaptive scripts (to automatically adjust the provided scaffolding to the needs of learners, e.g., fading the level of support or targeting increasingly more complex activities through support) and adaptable scripts (i.e., learners themselves discuss and decide which script components to include or to remove, to support self-regulation and metacognitive awareness).

3.3.1 Script-based argumentation learning

A number of approaches exist to improve the quality of argumentation through structured communication interfaces that implement or support specific pedagogical communication models. Sometimes, such approaches are referred to as *micro scripts* (Dillenbourg and Hong 2008). In terms of the script theory of guidance (Fischer et al. 2013; see above), they provide scaffolds operating at the scriptlet level. Some approaches try to encourage desired types of messages (e.g., arguments, evidence); other approaches provide message templates designed according to standards of good argumentation (e.g., messages consisting of a claim and reasons rather than bare claims); other approaches again try to promote fruitful interaction patterns (e.g., message sequences of the form argument-counterargument-integration). The overarching goal of such approaches typically is that students gradually internalize behaviors represented in the script and transfer these behaviors to situations in which the script is not available anymore.

Often, it is assumed that by improving the quality of argumentation, students also benefit in terms of increased domain knowledge learning. High-quality argumentation is transactive in nature since it involves well-reasoned, critical and constructive responses to learning partners. Therefore, according to the transactivity principle of the script theory of guidance (Fischer et al. 2013), promoting argumentation practices is a promising approach to foster the acquisition of domain knowledge. Approaches operating under this premise have been discussed in section 2.7.3 under the label *arguing to learn*. The kind of learning involved in such approaches is sometimes referred to as *argumentative knowledge construction* (Stegmann et al. 2007, 2012; Weinberger and Fischer 2006; Weinberger et al. 2010).



Michael is attributing internal stable

He says that he is not gifted and an attribution on own giftedness is seen as internal and stable by Weiner's theory

Perhaps he didn't tell the truth in this counseling session and he knows that he is onyl lazy.

1.
Claim:
Michael's parents are attributing external stable
Grounds:
They say they are not gifted and therefore Michael is not gifted, too. An attribution on giftedness is seen as internal and stable by Weiner's theory.
Qualification:
Perhaps the parents do not believe in this, but try to be insightfully

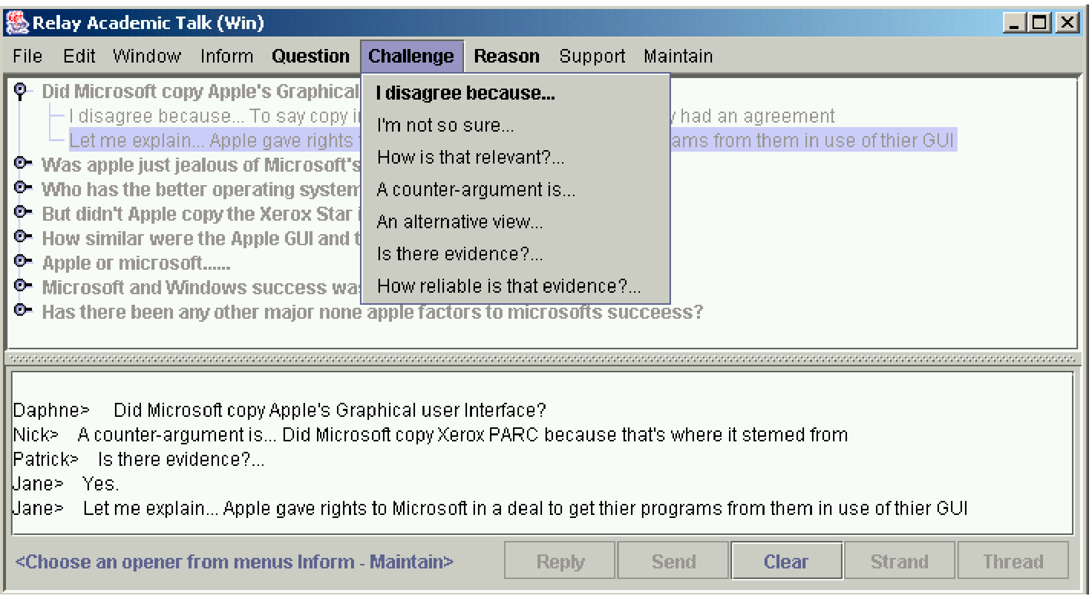
Figure 9: Form-like interface to scaffold the creation of individual messages in a forum discussion. From Weinberger et al. (2010).

Figure 9 shows an approach used in a series of studies (Stegmann et al. 2007, 2012; Weinberger et al. 2010) to scaffold the creation of messages according to a simplified version of the Toulmin model of argumentation (Toulmin 1958; see section 1.4.1). An argument template is provided with fields to enter a claim, grounds to support that claim, and qualifications to specify exceptional conditions under which the claim cannot be maintained. Empirical studies indeed confirmed that this pre-structuring of messages improves the formal quality of argumentation (i.e., fewer bare claims and more supported and qualified claims).

Maybe the most widespread approach to scripting argumentation is the use of *sentence openers* (or *note starters*). Sentence openers are predefined phrases students choose from to start new messages (Soller 2001). Typically, students complete these messages in their own words, but in some cases students also have to choose from a limited set of propositions to complete the message (Baker and Lund 1997). A number of systems enhance chat and threaded discussion interfaces with sentence openers, e.g., Group Leader Tutor (McManus and Aiken 1995), C-CHENE (Baker and Lund 1997), BetterBlether (Robertson et al. 1998), AcademicTalk (McAlister et al. 2004), InterLoc (Ravenscroft 2007), and the Future Learning Environment (FLE3; Oh and Jonassen 2007). Figure 10 shows the sentence opener interface of AcademicTalk (McAlister et al. 2004). Here, students select sentence openers from a set of menus, each containing a specific category of sentence openers (e.g., *inform*, *question*, or *challenge*). Other systems represent sentence openers as buttons in the user interface (e.g., Baker and Lund 1997). Weinberger et al. (2005; first study) prepared the input field of a discussion board with predefined sentence openers, which corresponded to specific roles. For instance, one of the sentence openers inserted for the constructive critic role was: *My proposal for an adjustment of the analysis is* ... That is, sentence openers were used to prompt students to think about and include certain aspects in a message rather than as options for initiating new messages. In this respect, the approach is more alike to the approach depicted in Figure 9, p. 112, than to the other sentence opener approaches discussed before.

Depending on the specific type of dialogue that researchers were hoping to foster, different sets of sentence openers have been used. InterLoc (Ravenscroft 2007), for instance, can support multiple types of dialogue through corresponding sentence openers, among others, critical discussions and creative reasoning dialogues. The dialogues are modeled as dialogue games and formalized in terms of participant roles, dialogue moves, corresponding sentence openers, and rules of interactions (i.e., how to best respond to specific dialogue move).

Since sentence openers can be mapped to specific communicative intentions, computers can achieve some level of dialogue understanding without complex natural language processing (Baker and Lund 1997). AcademicTalk (McAlister et al. 2004) and InterLoc (Ravenscroft 2007) capitalize on this by recommending sentence openers appropriate to respond to previous contributions, based on the rules of interaction of the underlying dialogue game (see the sentence opener highlighted in bold text in Figure 10). The Group Leader Tutor (McManus and Aiken 1995) uses sentence openers to support and diagnose specific collaboration skills (e.g., leadership, creative conflict) in the context of collaborative problem solving, based on a model that associates sentence openers with these collaboration skills (Johnson and Johnson 1991).



Did Microsoft copy Apple's Graphical
I disagree because... To say copy i … had an agreement
Let me explain... Apple gave rights … ams from them in use of thier GUI
Was apple just jealous of Microsoft's
Who has the better operating system
But didn't Apple copy the Xerox Star i
How similar were the Apple GUI and t
Apple or microsoft......
Microsoft and Windows success wa
Has there been any other major none apple factors to microsofts success?

Daphne> Did Microsoft copy Apple's Graphical user Interface?
Nick> A counter-argument is... Did Microsoft copy Xerox PARC because that's where it stemed from
Patrick> Is there evidence?...
Jane> Yes.
Jane> Let me explain... Apple gave rights to Microsoft in a deal to get thier programs from them in use of thier GUI

Figure 10: Sentence opener interface of the AcademicTalk system (McAlister et al. 2004). Adapted from McAlister (2004).

While sentence openers have the potential to shape student interactions in favorable ways, there are limitations and challenges that also must be considered. On the one hand, sentence openers can reduce students' typing load since frequently used text fragments can be added with a click of a button (Baker and Lund 1997; Lazonder et al. 2003; Soller 2001). On the other hand, sentence openers must be carefully and systematically organized in the user interface, possibly grouped according to higher-level categories, to help students quickly find appropriate sentence openers (Baker and Lund 1997; Soller 2001). The set of available sentence openers should be broad enough to satisfy students' communicative needs and avoid misuse (Soller 2001). Yet, too many options again reduce the salience of individual sentence openers and increase search time. Students may avoid the difficulty of classifying the content of their messages by always picking an unspecific sentence opener such as *I think ...* (Lazonder et al. 2003) One way to reduce the use of overly general or inappropriate sentence openers is to allow free-text messages in addition to scripted ones (Baker and Lund 1997). Yet, this has the potential to undercut the goals of the script as well, when students consistently ignore the provided sentence openers and use the interface like a standard chat (Lazonder et al. 2003).

While to date, relatively little research regarding the pedagogical effectiveness of sentence opener approaches has been published, the existing evidence clearly indicates that sentence openers can bias discussion behaviors to the better. For instance, it has been found that sentence openers improve task focus (Baker and Lund 1997; McAlister et al. 2004) and reflectiveness (Baker and Lund 1997) in discussions. With respect to argumentation, sentence openers have been shown to encourage critical engagement with the opinions of others (McAlister et al. 2004; Nussbaum et al. 2002) as well as the use of evidence and reasons to support claims (McAlister et al. 2004; Oh and Jonassen 2007), two important skills of argumentation many people do not make use of (Kuhn 1991; Weinberger and Fischer 2006). Similar results are reported for other approaches based on form-like user interfaces (see Figure 9, p. 112; Stegmann et al. 2007, 2012; Weinberger et al. 2010). The results with respect to domain knowledge acquisition are less clear including positive results (Weinberger et al. 2005; Weinberger et al. 2010) but also some null results (Oh and Jonassen 2007; Stegmann et al. 2007, 2012). That is, process improvement did not always lead to domain knowledge gains.

Often, it is not sufficient to provide students with basic communication and collaboration tools even if these tools are well designed (Dillenbourg et al. 1996). Based on this insight, researchers and practitioners have tried a number of approaches to make argumentation more successful, including the provision of relevant background information, tool familiarization, and procedural instructions. To make argumentation a more situated activity, some have contextualized the use of argumentation tools by designing wider curricula: For instance, SenseMaker (see Figure 8, p. 106) is part of the Web-based Inquiry Science Environment (WISE), for which a number of curriculum projects have been designed (Linn et al. 2003); Suthers et al. (1997) developed activity plans, problems with accompanying Web-based materials, and assessment instruments that could be used, together with Belvedere (see Figure 7, p. 103), to implement scientific inquiry in the classroom.

Pedagogical models, or instructional plans, that particularly focus on structuring the collaboration process on a macroscopic level are sometimes referred to as *macro scripts* (Dillenbourg and Hong 2008). In terms of the script theory of guidance (Fischer et al. 2013), macro scripts provide scaffolds operating at the play, scene, and role level. Several macro-scripting approaches have been used to foster argumentation. For instance, students possibly collaborate and learn better when they first prepare themselves individually before joining a group discussion (Baker 2003; Schwarz and Glassner 2007), when they receive different background materials to make collaboration necessary for an optimal solution (e.g., Suthers et al. 2008), and when they have been assigned different roles to distribute tasks and emphasize the particular responsibilities of the individual (e.g., Nussbaum et al. 2007; Schellens et al. 2007; Weinberger et al. 2005).

Some approaches acknowledge the dialectical character of argumentation, that is, the origin and motivation for argumentation should be a conflict of opinion that provides a reason for argumentation, otherwise argumentation might become aimless (Van Eemeren and Grootendorst 2004); learning happens then by resolving this socio-cognitive conflict (Doise and Mugny 1984). There are different strategies that can create or maximize such conflicts artificially, for instance, by grouping students with different a priori opinions (Baker 2003) or assigning roles that represent different, opposite opinions in a role play scenario, which can be further amplified by preparing students with different background materials according to their roles (Muller Mirza et al. 2007). The literature shows that both approaches—capitalizing on existing opinion differences and creating artificial ones—can be effective. For instance, several studies show that composing student groups in a way that maximizes opinion conflicts can have positive effects on the quality of student discussions (Clark et al. 2009; Jermann and Dillenbourg 2003). Weinberger et al. (2005; study 1) found that assigning the roles of a case analyst and a constructive critic to students was beneficial in terms of individual knowledge acquisition.

Macro scripts may be realized through paper or oral instructions but can also be supported through technology. For instance, some computer-based tools support instructors in planning and defining macro scripts. Figure 11 shows the planning tool of the Metafora platform (Dragon et al. 2013). The overarching goal of the Metafora project is to support students in *learning to learn together*. Students, rather than instructors, use the tool to define, monitor, and reflect on their learning plans to tackle open-ended inquiry challenges in math and science. Argumentation is one of the central activities students engage in, e.g., when discussing conflicting interpretations of results obtained in experiments or conflicting approaches to tackle a problem. The visual language provided with the planning tool allows the definition of activity stages, processes, resources, attitudes, and roles, represented as cards in a shared planning space. Learning tools and resources can be accessed directly from the created plan.

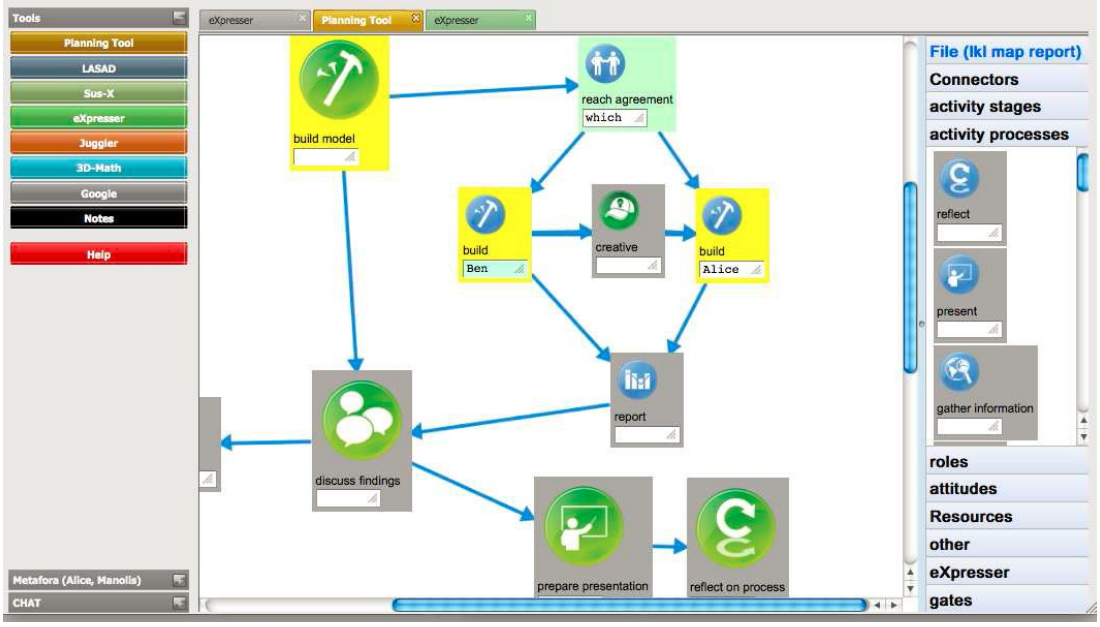


Figure 11: Planning tool of the Metafora system (Dragon et al. 2013). From Mavrikis et al. (2012).

An important research challenge is how to operationalize collaboration scripts (Tchounikine 2008), that is, to devise script languages to formally describe collaboration scripts, and to design script engines to interpret and execute such formal description in computer-based learning environments. Another important question is how to enhance collaboration scripts with elements of adaptivity (Harrer et al. 2008; Tsovaltzi et al. 2010). As discussed, current efforts in pedagogical research aim at building a theoretical foundation for scripting approaches (Fischer et al. 2013). Rather than focusing on the effects of individual scripts and script elements, such a theory aims at describing in more general terms how external scripts are transformed into internal scripts (i.e., cognitive structures in the minds of learners). An important consideration here is *over-scripting* (Dillenbourg 2002), that is, the external structure may hinder effective collaboration rather than help. For instance, this may happen when the external scripts conflicts with an already existing internal script or when the external script has already been internalized, in which case the external structuring causes unnecessary (extraneous) cognitive load. A possible solution is the above mentioned adaptation mechanisms, which allow tailoring the nature and amount of structuring to the actual needs of learners. For instance, to avoid extraneous cognitive load, an adaptive script may gradually reduce the amount of structuring as learners internalize the script and acquire the competency to self-organize their learning (*fading the scaffold*; cf. Pea 2004).

3.4 Automated Analysis of Argumentation

Bell (1997) makes a distinction between *argument modeling systems*, which support the analysis and structural representation of arguments (e.g., using diagrams), and *discussion-oriented systems*, which provide a medium for argumentative exchange between discussants (e.g., a sentence opener interface). While discussion-oriented systems often aim at a broad set of communication and collaboration skills, such as balanced participation, topic focus, and leadership, argument modeling systems focus on the logic of arguments and domain-specific argument structures. For instance, scientific argumentation requires that hypotheses are interrelated with data in order to check the amount of empirical support and opposition. An argument modeling system can check whether a sufficient amount of such relations has been considered in an argument model (e.g., by counting the number of corresponding links in an argument diagram). Given the different analysis focuses and setups, the two system classes employ different analysis approaches.

Analysis approaches in argument modeling systems can typically capitalize on well-structured, thus more easily interpretable argument representations created by users (e.g., a diagram with labeled nodes and links). The following approaches can be distinguished:

- *Syntactic analyses* (e.g., Pinkwart et al. 2009; Suthers et al. 2001) check whether created argument representations comply with a set of given syntactic constraints (e.g., data supports hypotheses and not vice versa). Typically, such constraints are defined by domain experts and implemented using logic or rule-based programs (e.g., Prolog or Jess rules).
- *Problem-specific analyses* (e.g., Dragon et al. 2006; Pinkwart et al. 2009; Suthers et al. 2001) check whether the created argument representation adequately models a given problem case. This is typically achieved by comparing student solutions with expert solutions to the same problem. Since in many cases, the modeling of arguments is an ill-defined problem with many possible solutions, heuristics must be used to identify deviations from the expert model.

- *Simulations of reasoning / decision-making processes* (e.g., Gordon et al. 2007; Ranney and Schank 1998) determine whether a claim is believable / acceptable based on the created argument representation. For instance, in the Ranney and Schank (1998) approach, students define the structure of an argument in terms of elements (hypotheses, evidence), and relations (e.g., a piece of evidence supports a hypothesis, two hypotheses contradict one another). Based on additional student-provided parameters (e.g., strength of evidence, weights of relations), a connectionist model can compute believability scores for hypotheses. The Gordon et al. (2007) approach is based on a logical formalism with rules that operationalize specific proof standards.

- *Assessments of content quality* (e.g., Pinkwart et al. 2009) determine the quality of the textual content of individual argument components. Since a fully automated approach is technically hard to realize, Pinkwart et al. (2009) utilized collaborative filtering techniques (Goldberg et al. 1992), that is, an overall assessment is computed by aggregating ratings of peer learners.

- *Classifications of the current modeling phase* (e.g., Pinkwart et al. 2009; Suthers et al. 2001) determine whether the student is, for instance, in an orientation, modeling, or reflection phase (i.e., problem solving is conceived of as a multi-phase process). For instance, certain characteristics of the current diagram version may be heuristically interpreted to determine the current phase (e.g., a students is probably in an early phase when the diagram contains only very few elements).

Analysis approaches in discussion-oriented systems have to cope with natural language text input to a greater extent, which still poses a stiff challenge for a computational analysis. Some approaches can benefit from more structured communication interfaces, e.g., sentence openers (Soller 2004) and graphical discussions (McLaren et al. 2010). The following approaches can be distinguished:

- *Analyses of process characteristics* (e.g., Rosé et al. 2008) identify the dialogue function of discussion moves and speaker intentions, for instance, counterarguments and question-answer interactions in dialogues.

- *Analyses of discussion topics* (e.g., Goodman et al. 2005; Kim et al. 2008) identify the topics covered in discussions.

- *Analyses of interaction problems* (e.g., Goodman et al. 2005; Soller 2004) identify, for instance, unanswered questions and failed attempts to share knowledge.

- *Assessments of collaboration quality of longer sequences of time* (e.g., Goodman et al. 2005) aggregate and summarize students' behaviors over time, for instance, the level of group responsiveness and agreement.
- *Classifications of the current discussion phase* (e.g., Israel and Aiken 2007) determine whether the group is, for instance, in a confrontation, opening, argumentation, or conclusion phase (i.e., a discussion is conceived of as a process that unfolds into multiple phases).

Technically, a wide range of techniques is used to develop such analysis approaches, including machine learning (Mu et al. 2012; Rosé et al. 2008; Soller 2004), information retrieval (e.g., TF-IDF based similarity to topic vectors [Kim et al. 2008]), and manually defined heuristic rules and models (Israel and Aiken 2007).

3.5 Adaptive Support of Argumentation

Automated argument / discussion analysis is not an end in itself; it typically serves the purpose of generating adaptive support for students (and other stakeholders, e.g., discussion moderators). A computational analysis allows the system to identify salient features of the learning process. Mechanisms of adaptation can then tailor guidance and support to the specific needs and problems of learners, e.g., textual messages that hint at possible errors in an argument diagram, or highlighting of specific sentence openers to recommend a particularly useful response type. Another example for a possible adaptation is *fading the scaffold*, that is, reducing the level of support based on the progress learner make. Rigorous empirical investigations of adaptive support for collaborative or argumentative learning are still rare—some of the few results available will be discussed in Part C. The presentation below focuses on a categorization of adaptive support approaches and corresponding examples. In particular, support approaches will be discussed along three dimensions: mode and content, timing, and selection strategies.

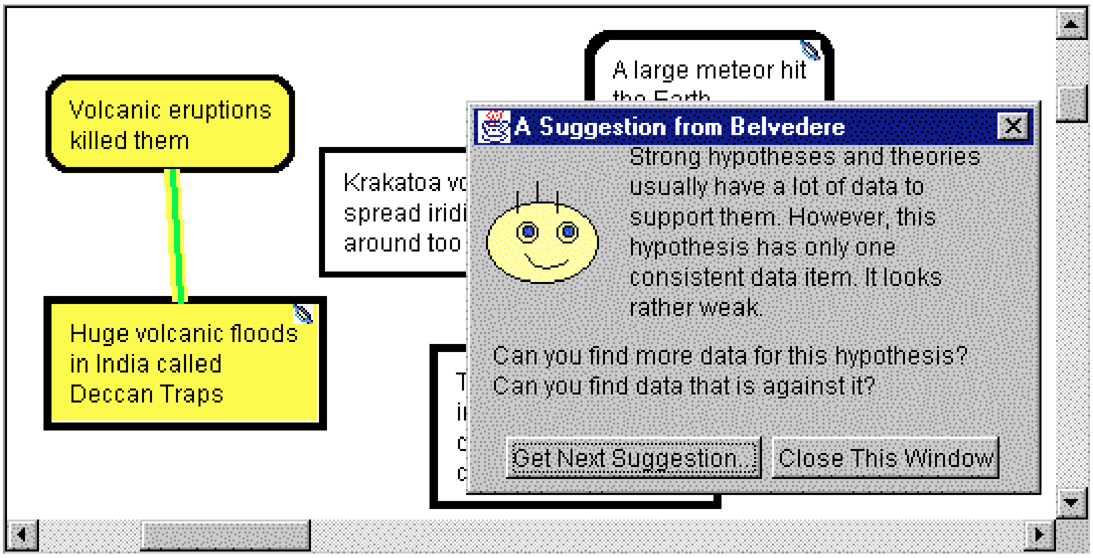
3.5.1 Mode and content

Adaptive argumentation learning systems take the following approaches to present feedback to students or other stakeholders:

- *Textual feedback* presented to the student is certainly the most common form of support. Belvedere (Suthers et al. 2001) and Rashi (Dragon et al. 2006) use textual messages to foster inquiry skills and the learning of principles of scientific argumentation (e.g., hypotheses should explain observed data);

LARGO (Pinkwart et al. 2009) supports the analysis of a legal transcript by presenting short versions of the five most relevant feedback messages to the student. Feedback in these systems is based on pre-canned text messages, and formulated as suggestions / prompts for self-reflection rather than imperative / corrective formulations. The reason is to avoid confusion when a diagnosis is a "false alarm" and to foster the development of the students' skills of self and peer critiquing, i.e., the feedback should encourage the student to think for him or herself about the diagram and possible weaknesses (Suthers et al. 2001). While the just discussed systems provide feedback to support the creation of argument diagrams, other systems aim at promoting productive group discussions (Goodman et al. 2005; Israel and Aiken 2007). The dialogue strategy may also target the elicitation of correct conceptual knowledge rather than a proper use of arguments (Kumar et al. 2007). Table 18 shows example feedback messages used in some prominent argumentation modeling and discussion-oriented system.

- *Highlighting* of relevant portions of an argument diagram may help students easily identify parts of the solution they need to pay special attention to. Typically, highlighting is then provided together with some textual message that explains what to do or what is wrong with the highlighted portion of the diagram (Pinkwart et al. 2009; Suthers et al. 2001). Figure 12 shows a screenshot of Belvedere, in which relevant parts of the diagram are highlighted in yellow, accompanied by a textual message. Instead of highlighting parts of an argument diagram, systems may also highlight other elements in the user interface, for instance, specific sentence openers to indicate recommended reply types to previous messages (McAlister et al. 2004; see Figure 10, p. 114, menu items in bold text).
- *Meters* are sometimes used to display group indicators (e.g., dialogue speed, relative amount of statements needing a reply) and student indicators (e.g., certainty level, activity level) to support cognitive / process awareness and meta-cognition (e.g., Dragon et al. 2006; Goodman et al. 2005). Meters might also be used to support teachers and moderators. Figure 13 shows a screenshot of the Moderator's Interface developed in the Argunaut project (McLaren et al. 2010). The interface visualizes awareness information to support moderators of graphical educational debates. The panel on the left shows the list of currently ongoing debates and students who joined in to the debate (green – active debates / users; gray – inactive debates / users). The panel in the middle shows



Volcanic eruptions killed them

Huge volcanic floods in India called Deccan Traps

Krakatoa vo spread iridi around too

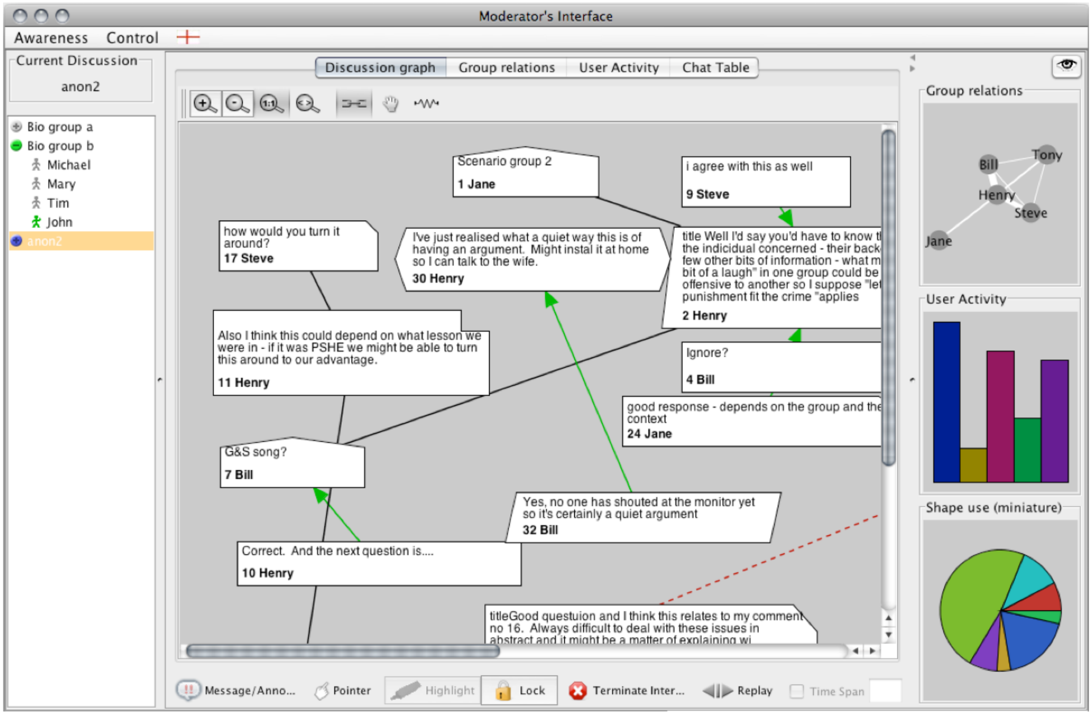
A large meteor hit the Earth

A Suggestion from Belvedere

Strong hypotheses and theories usually have a lot of data to support them. However, this hypothesis has only one consistent data item. It looks rather weak.

Can you find more data for this hypothesis? Can you find data that is against it?

Figure 12: A feedback message in the Belvedere system (Suthers et al. 2001). From Suthers et al. (2001).



Scenario group 2
1 Jane

i agree with this as well
9 Steve

how would you turn it around?
17 Steve

I've just realised what a quiet way this is of having an argument. Might instal it at home so I can talk to the wife.
30 Henry

title Well I'd say you'd have to know t the indicidual concerned - their back few other bits of information - what m bit of a laugh" in one group could be offensive to another so I suppose "let punishment fit the crime "applies
2 Henry

Also I think this could depend on what lesson we were in - if it was PSHE we might be able to turn this around to our advantage.
11 Henry

Ignore?
4 Bill

good response - depends on the group and the context
24 Jane

G&S song?
7 Bill

Yes, no one has shouted at the monitor yet so it's certainly a quiet argument
32 Bill

Correct. And the next question is....
10 Henry

titleGood questuion and I think this relates to my comment no 16. Always difficult to deal with these issues in abstract and it might be a matter of explaining wi

Figure 13: Awareness information displayed in Argunaut's Moderator's Interface (McLaren et al. 2010).

Table 18

Exemplary textual feedback messages in different argumentation systems

System	Example message purpose	Example message content
Belvedere (scientific inquiry; Suthers et al. 2001)	Avoid confirmation bias	"You've done a nice job of finding data that is consistent with this hypothesis. However, in science we must consider whether there is any evidence *against* our hypothesis as well as evidence for it. Otherwise we risk fooling ourselves into believing a false hypothesis. Is there any evidence against this hypothesis?"
	Discriminate between alternative hypotheses based on (especially negative) evidence	"These hypotheses are supported by the same data. When this happens, scientists look for more data as a 'tie breaker' - especially data that is *against* one hypothesis. Can you produce some data that would 'rule out' one of the hypotheses?"
Rashi (scientific inquiry; Dragon et al. 2006)	Build bottom-up arguments (i.e., arguments from data to hypotheses)	"Here's a list of possible arguments. Try to pick the one you can support or refute with the data you have already: <*list of arguments*>"
	Repair wrong relationship type between propositions (student solution different from ideal expert model)	"Are you satisfied with the relationship you have established between *P1* and *P2*?" *(student can select between "Yes, it is correct" and "No, help me to fix the relationship")*
LARGO (analysis of legal argument transcripts; Pinkwart et al. 2009)	Repair modeling weakness: Hypothetical elements does not relate to Test element	"In your solution, the hypotheticals H1 and H2 are distinguished from each other. Yet, hypothetical H2 is not related to any test or the current fact situation. Please explain why you did so, either in free text or by modifying the diagram."
	Consider important transcript passage that have not been considered yet	"Please look at this part of the transcript (scroll to line *L*) and explain its role within the argument."
Group Leader Tutor (group deliberation; Israel and Aiken 2007)	Avoid off-topic contributions	"Please try to stay on-topic while working with your group"
	Express only one idea per messages	"You are trying to express several ideas in one sentence. Please re-enter your statements, one idea at a time."
EPSILON / Pierce (group deliberation; Goodman et al. 2005)	Respond to peer messages that have not yet been answered	"Sarah said '…'. What do you think about that, Jeremy?"
	Elicit help from peers when something has not been understood	"Excuse me, Milhouse, but I think you might be confused. You should ask for help on this topic."

Note: If messages refer to diagram elements those elements are typically highlighted in the diagram. Adapted from Scheuer et al. (2012).

one selected debate in detail. The panels on the right visually summarize important aspect of the selected debate, in particular, group relations (i.e., who communicates with whom how frequently; top panel), user activity (e.g., amount of create, modify, and delete actions; middle panel), and shape use (i.e., proportion of the different node and link types in the diagram; bottom panel). Other functionalities include applying analysis rules that automatically identify important aspects in debates through shallow analysis techniques (e.g., occurrences of predefined keywords) and artificial intelligence techniques (e.g., classification of off-topic contributions).

3.5.2 Feedback control and timing

Adaptive argumentation learning systems take the following approaches regarding when to present feedback to students:

- *On-demand feedback* is provided only upon a student's request (e.g., Dragon et al. 2006; Pinkwart et al. 2009; Suthers et al. 2001). There are several reasons why such a strategy may be beneficial: First, the feedback is provided when the student really wants it, not interrupting naturally occurring activities (Dragon et al. 2006). Second, the student is not flooded with unnecessary messages since he or she decides the feedback frequency. Third, the construction of an argument diagram is a continuous process, with sometimes no clear end or conclusion, hence it makes sense to let the user decide when the process is ready to be checked (Pinkwart et al. 2006). Fourth, on-demand feedback allows the student to assume more control and the tutoring component less control, possibly leading to more student motivation and less student discouragement (Suthers et al. 2001). On the downside, some students take minimal or no advantage of on-demand feedback, even when they are stuck and obviously need assistance (Pinkwart et al. 2009; Suthers et al. 2001).
- *Immediate system feedback* is provided right after a mistake or problem is identified, without a student explicitly requesting help (Goodman et al. 2005; Israel and Aiken 2007; Kumar et al. 2007). Especially when feedback is intended to scaffold and improve the current student activity, it may be best provided immediately. For instance, when a discussion is drifting off-topic, immediate feedback can be used to re-focus students again. Furthermore, as mentioned above, many students do not make use of on-demand feedback and thus miss learning opportunities. One reason might be that students are not aware of their suboptimal behavior / solution. On the downside, if the amount

of feedback becomes excessive, it could distract the student. Moreover, immediate feedback might be perceived as annoying or disruptive to the students' natural flow of interaction.

- *Summative system feedback* is provided after a session has finished. Typically, the goal is to give students the opportunity to reflect on their activities (Israel and Aiken 2007). A positive aspect is that delayed feedback does not interfere with on-going students' activities. However, such feedback is not able to scaffold the student activities in the context in which a problem occurs. Intelligent tutoring system research yielded mixed results with respect to whether immediate or delayed feedback approaches are more effective (Shute 2008).

3.5.3 Feedback selection and priority

It is often required to control the frequency of feedback in order to avoid excessive amounts of messages. The system may have to decide which messages are most relevant in the current situation. Belvedere (Suthers et al. 2001) and LARGO (Pinkwart et al. 2009) provide the most important and short versions of the five most important feedback messages, respectively, when students request help. Criteria to be considered in the prioritization of messages are: current problem-solving phase, age of diagram structures the advice is referring to, type of advice, and requestor of advice.

3.6 Summary

A variety of new computer-based technologies to support the learning and practical use of argumentation emerged during the last few decades. Throughout the literature one can find at least three principal approaches that have been taken to build argumentation tools and which have partly been explored as part of this dissertation. Table 19, p. 128, summarizes and contrasts the differences between these three high-level approaches.

- *Representational guidance approaches* (e.g., Nussbaum et al. 2007; Pinkwart et al. 2009; Suthers and Hundhausen 2003) provide external, oftentimes modifiable representations of argumentation structures with the aim of stimulating and improving individual reasoning, collaboration, and ultimately learning. A theoretical underpinning of such approaches is provided with the theory of representational guidance (Suthers 2003). The probably most widespread approach is to represent arguments in the form of node-and-link graphs.

- *Discussion scripting approaches* provide structuring elements for argumentation learning processes with the aim of fostering rich and high-quality interactions. A theoretical underpinning of such approaches is provided with the script theory of guidance (Fischer et al. 2013). Discussion scripts may operate directly at the level of individual discussion moves, an approach sometimes referred to as *micro scripting* (e.g., McAlister et al. 2004; Schwarz and Glassner 2007; Soller 2001; Stegmann et al. 2007). Typically, micro scripts are realized through special-purpose communication interfaces that encourage (or sometimes force) a desired mode of interaction. One of the main approaches is to let students choose between predefined sentence openers when composing new text messages. Discussion scripts may also provide more coarse-grained structures by defining phases, roles, and activities, an approach typically referred to as *macro scripting* (Dillenbourg and Hong 2008; Lund et al. 2007; Schellens et al. 2007).
- *Adaptive support approaches* (e.g., Pinkwart et al. 2009; Suthers et al. 2001) aim at more dynamic forms of help by providing pedagogical feedback on student actions and solutions, hints and recommendations to encourage and guide future activities, or automated evaluation services to indicate whether an argument is in its current form acceptable or not. Here, techniques from the research fields of artificial intelligence and intelligent tutoring systems are widely used. Two different tasks are important to provide adaptive support: (a) analyzing student actions, interactions, and created artifacts to obtain some level of machine understanding of the student, the group, and the process, and (b) generating appropriate support based on pedagogically informed support strategies. Two main approaches to developing automated analysis mechanisms are expert knowledge modeling and machine learning. Rigorous empirical research with respect to adaptation strategies [part (b)] is almost absent; a broad and solid theoretical underpinning, or *theory of adaptation* for collaborative and argumentative learning, still lacking. One reason is certainly the complexity inherent to part (a).

The research presented in this dissertation considers each of these three approaches sketched above. Chapter 4 investigates an approach that combines argument diagrams (i.e., a representational guidance approach), and sentence openers (i.e., a discussion scripting approach). The basic idea is that students first represent given texts as diagrams and then discuss these diagrams, in a structured way, using a chat enhanced with sentence openers. The sentence openers correspond with a role-based macro-script, which assigns to each student two roles: proponent of the own position and a critic of the partner's position. For instance, to support the role of a critic, sentence openers are chosen to encourage questions, rebuttals, and critical statements. A second macro-script component guides students through four phases with predefined activities, starting with the creation of the diagrams and proceeding with three phases of a critical discussion (essentially, clarifying positions, arguing, and concluding). Chapter 5 and Chapter 6 are concerned with approaches to the automated analysis and adaptive support of argumentation learning. A conceptual and technical framework for generating adaptive support has been developed (Chapter 5) and two principal classes of methods to developing automated analysis functionality researched (one described in Chapter 5 and the other in Chapter 6). After having described the specific research, the final Part C provides a more in-depth discussion of the three approaches sketched here (i.e., representational guidance, collaboration scripting, and adaptive support) also considering the specific research results of this dissertation.

Table 19

Three main approaches to support argumentation learning: representational guidance, discussion scripting, and adaptive support

Approach	**Collaborative or individual learning?**	**Pedagogical rationale**	**Possible roles of technology**
Representational guidance	Collaborative and individual learning	Stimulating and improving individual reasoning and collaboration through explicit knowledge representations that reify concepts important in the specific argumentation domain	Editable knowledge representations in the user interface (e.g., argument diagrams)
Discussion scripting	Collaborative learning	*Micro-scripting:* Structuring communication at a microscopic level to encourage or force high-quality discussions *Macro-scripting:* Structuring the learning process at a macroscopic level (i.e., above the level of single communication moves or sequences) through pedagogically defined student roles, phases and activities	*Micro-scripting:* Structured communication interface (e.g., users choose between message categories or sentence openers) *Macro-scripting:* System-controlled access to tools and resources based on a formal and machine-readable definition of the learning process (phases, activities, and user roles)
Adaptive support	Collaborative and individual learning	Adapting the learning environment to individual and dynamically evolving aspects, including characteristics of the individual learner, the learning group, as well as the current state and history of the problem-solving / collaboration process	Automated analysis of the learner, the group, created artifacts, and the learning process Generation of adaptive support based on pedagogically informed strategies

Part B

Research Components

Chapter 4
Supporting Argumentation Learning through Multilevel Scaffolding

This chapter discusses a novel user interface approach to supporting critical, elaborative discussions between students. The present approach brings together two separate strands of past research, namely, supporting argumentation learning through argument diagrams (background in section 3.2) and supporting argumentation learning through collaboration scripts (background in section 3.3). The motivation for combining both methods is to capitalize on the strengths of each method; in the best of all cases, the combination may even result in positive synergistic effects on performance and learning.

This chapter discusses the rationale for the new approach and formulates a corresponding research question (section 4.1), describes the specific elements of the approach (section 4.2), and presents a study that provides partial confirmation for the underlying research hypotheses (section 4.3 and section 4.4). Section 4.5 summarizes and discusses the research presented in this chapter.

4.1 Research Question

With an improved understanding of individual scaffolding approaches, research is becoming increasingly interested in the question of how to combine different approaches in fruitful ways (Schellens and Fischer 2013; Tabak 2004). Argumentation diagramming and discussion scripts are two methods successfully used in the past to support argumentation learning. Argumentation diagramming has been shown to help students construct, reconstruct, and reflect on arguments. However, while diagrams can serve as valuable resources—or even guides—during conversations (Suthers and Hundhausen 2003), they do not provide explicit and immediate support for the discussion itself. Conversely, discussion scripts can provide direct support for the discussion, e.g., through sentence openers that encourage high quality discussion moves (e.g., Nussbaum et al. 2002). Yet, students often struggle to comply with the rules of a script, as evidenced by both the misuse and nonuse of sentence openers (Lazonder et al. 2003; Soller 2001). A possible reason is that students lack a sufficient understanding of the relevant *space of debate* (Baker et al. 2007), that is, the viewpoints, values, standards of reasoning, facts, claims, and arguments relevant to a discussion domain. Unexperienced discussants may be overwhelmed with developing a proper understanding of the space of debate while, simultaneously, being tasked with engaging in high-quality forms of collaborative argumentation they are not familiar with either. More experienced discussants, on the other hand, may be able to develop a proper understanding of the space of debate on the fly *through* participation in collaborative argumentation.

Essentially, each method provides unique support on one specific level: Argument diagrams support students in understanding the content of a space of debate by structuring a body of domain knowledge in terms of relevant claims, facts, and arguments. Thus, diagrams provide direct support at the content level and may therefore be considered primarily as *content scaffolds*. Discussion scripts, on the other hand, support students in engaging in desired modes of student-to-student interaction, that is, support on the social level. Therefore, discussion scripts may be considered as primarily *social scaffolds* (Weinberger et al. 2005).

Both methods share an orientation towards *epistemic activities*, that is, activities aiming at the construction of knowledge and meaning. Argument diagrams may be considered as epistemic scaffolds since they explicitly represent "epistemological concepts" (Suthers 2003)—concepts that play an instrumental role in the construction of knowledge through argumentative reasoning, for instance, the concepts *claim*, *fact*, *argument*, *for*, and *against*. The reification of epistemological concepts may help students better understand these concepts themselves as well as the specific content classified and structured in the diagrams according to these concepts. Discussion scripts may be considered as epistemic scaffolds, too, since they aim at the co-construction of knowledge—a mode of knowledge construction that is rooted in social interaction. Discussion script elements like sentence openers also reify epistemological concepts, albeit in a somewhat less explicit form (e.g., a sentence opener *According to a statistic / estimate* may be associated with categories such as *fact* or *evidence*).

Overall, each method—diagramming and scripting—provides unique support by focusing on one specific aspect: content and social interaction, respectively. Yet, both methods may also be conceived of as epistemic scaffolds since they explicitly support epistemic activities. This shared orientation may function as a powerful bridge between both types of scaffolding. For instance, Tabak (2004) identified "cohesion and direct interaction between the elements of a scaffolding system" as potential conditions to achieving synergistic scaffolding effects.

In summary, since the generation of high-quality discussion moves requires both, familiarity with the space of debate (i.e., content) and appropriate forms of interpersonal exchange (i.e., social interaction), students may benefit from being supported on both levels. Hence, by combining both of these instructional techniques, it may be possible to capitalize on their advantages, while minimizing their disadvantages. This chapter will address the following research question:

> **(RQ1) Multilevel Scaffolding:** "Does a user interface that integrates argument diagramming with a discussion script promotes the quality of student-to-student interaction and content learning more than each individual method?"

This strand of research also contributes to the question of how to best arrange the educational use of multiple representational formats (cf. Ainsworth 1999), in particular, representing arguments both in a diagram and in a text chat to support learning. Prior research in this direction investigated, for instance, the effects of creating diagrams individually before a debate versus collaboratively during a debate in a text chat (Munneke et al. 2003), the effects of using diagrams as discussion medium as opposed to using them to represent a preceding text chat discussion (Lund et al. 2007), and ways to integrate diagrammatic and typewritten discourse (Suthers et al. 2008).

4.2 Approach

To investigate research question RQ1, a LASAD configuration that combines an argument diagramming interface and a sentence opener interface was set up (more on LASAD in section 5.1) and embedded within a discussion script to promote critical, elaborative discussions based on the roles of a proponent and a constructive critic. The theoretical foundation of this *peer critique approach* is given by the socio-cognitive conflict theory (Doise and Mugny 1984; see section 2.6.3), which postulates that the attempt to resolve social conflicts is a key impetus of cognitive development and learning. Being confronted with conflicting opinions triggers processes that are conducive to learning, such as explaining, justifying, refining, and integrating knowledge (Nussbaum 2008). Similar tactics aimed at inducing and emphasizing conflict to promote discussions and learning have been used in domains such as physics (Baker 2003; Clark et al. 2009) and instructional design (Jermann and Dillenbourg 2003; Weinberger et al. 2005).

Table 20
Sentence openers used as part of the discussion script to support the roles of a proponent and a constructive critic

Proponent	**Constructive Critic**
A central point is …	I would phrase this differently: …
An argument for this point is …	An argument against this point is …
For instance, …	But …
According to a statistic / estimate …	Do you have any / more evidence …
Do you have questions regarding …	Could you explain to me …

Note: Translated from German. Adapted from Scheuer et al. (2014).

Table 20 shows the sentence openers used in the discussion interface, which were exactly defined to support the roles of a proponent and a constructive critic. The specific openers were chosen to cover main move categories commonly considered, and deemed important, in the literature (cf. Baker et al. 2007; Soller 2001; Weinberger and Fischer 2006). Other considerations included the nature of the specific texts to be discussed (e.g., many arguments in the texts employ statistics; more on the texts below) and the goal to keep the user interface easy to learn and use (therefore: focus on a small set of core moves, use of intuitive formulations). Proponent moves included: making claims (*A central point is ...*), supporting one's case with arguments and illustrative examples (*An argument for this point is ...*, *For instance ...*), backing arguments with hard facts and statistics (*According to a statistic / estimate ...*), and asking for possible understanding problems to promote positive peer interaction (*Do you have questions regarding ...*). Constructive critic moves included: proposing changes with respect to wording and content of peer statements (*I would phrase this differently: ...*), making counterarguments (*An argument against this point is ...*), raising doubts or objections (*But ...*), and eliciting evidence and elaborations (*Do you have any / more evidence ...*, *Could you explain to me ...*). The openers were tested in several small-scale trial runs. The decision which specific openers to use was ultimately based on the judgments of and discussions between the involved researchers. It should be noted that some of the openers were considerably less verbose in their German original version, and thus, potentially handier to use, for instance, the German versions of *An argument for this point is ...* (*Dafür spricht ...*) and *An argument against this point is ...* (*Dagegen spricht ...*).

Furthermore, the script utilizes two opposing texts to promote conflict (as proposed in Andriessen and Schwarz 2009) and to provide necessary background information to enable fruitful discussions (cf. Suthers et al. 2008). Students discuss these texts in pairs with the ultimate goal to agree on a well-reasoned and joint conclusion. Each student represents the position of one of the two texts playing the proponent role while arguing against the position in the other text playing the constructive critic role. Each student has direct access only to "his" text while learning about the other text through his partner. Hence, to achieve the best possible joint conclusion, students depend on the contribution of their learning partners. This condition of positive resource interdependence has been found to be beneficial for collaborative learning since it promotes cooperation and reduces solitary activities and competition (Johnson and Johnson 2009; see also section 2.6.2). On face value, the two objectives of promoting conflict and promoting cooperation seem to be at odds. Yet, the two objectives can be conciliated in that conflict is the starting point and impetus, while the productive cooperative resolution of this conflict is the ultimate goal. The script is therefore designed to support epistemic rather than relational (or personal) forms of conflict resolution (Darnon et al. 2007), that is, an attitude of "being critical of ideas, not people" (Johnson and Johnson 1994) is promoted.

In the role of the proponent, students present, explain, and justify positions and arguments of their own text. In the role of the constructive critic, they carefully attend to the argumentation of their partners. Equipped with facts and arguments learned from their text or derived from own background knowledge and reasoning, the critic tries to identify weaknesses, provides counterarguments and objections, and requests clarification where needed. No specific instructions are given regarding when to take which role, so students can freely decide, based on the current situation, whether proponent or constructive critic moves are appropriate. Typically, due to the oppositional positions presented in the texts, arguing against the partner's text ultimately results in presenting and defending elements of one's own text. For instance, to attack a claim of the partner, one may present arguments and facts of one's own text. As described below, the discussion process is furthermore subdivided into three phases each providing more specific instructions. The role script is abandoned in the last phase, in which students advocate their personal opinions and try to arrive at a joint conclusion.

In summary, the discussion script:

- aims at engaging student dyads in critical, elaborative discussions,
- gives each student exclusive access to one of two opposing texts,
- assigns to students the roles of a proponent of their text and constructive critic of their partner's text,
- supports these roles through corresponding sentence openers in the user interface, and
- defines the overall group goal of agreeing on a joint, well-reasoned conclusion.

Before students engage in collaborative discussions, they individually analyze their text by representing it in an argument diagram. Past research has shown that successful collaboration usually involves a combination of individual and collaborative activities (e.g., Baker 2003; Jermann and Dillenbourg 2003; Rummel and Spada 2005). Individual preparation gives students time to make up their own minds about a controversial issue. It allows students to develop their own ideas before the ideas of others influence their thinking. Thus, more diverse knowledge resources can be activated and contributed to collaborative argumentation (Weinberger et al. 2007). With a clear picture on a given topic in mind, gained from careful individual deliberation, students are better prepared to engage in fruitful interaction with others.

During the discussion, the argument diagram creates a shared focus on the argumentative structuring of the given texts (e.g., contained claims and arguments; cf. Suthers 2003). It thus contributes to an orientation towards epistemic rather than relational forms of conflict resolution. The argument ontology (i.e., the set of available node and link types) has been intentionally designed in a relatively simple and informal way, allowing students to quickly grasp how to use the tool. Some prior research has shown that complex representational schemes can lead to confusion on part of the students and are therefore often more detrimental than beneficial in guiding students' thinking and interactions (Suthers et al. 2001). The specific ontology includes four box types (*main thesis*, *main argument*, *helping argument*, and *fact*) and two link types (*support* and *opposition*). Important statements can be identified at a glance since judgments regarding the importance of statements are explicitly and visually represented (*main arguments* versus *helping arguments*).

To provide guidance and structure to the overall learning process, argument diagramming and scripted discussion activities have been embedded into a collaboration macro script that arranges the different activities in a well-defined sequence of activity phases, shown in Table 21. In phase 1, students individually read and analyze the given texts and represent the respective lines of argumentation in a diagram. In phase 2, students discuss, based on the diagrams, aspects of the individual texts with their partner. In phase 3, students discuss relationships between the two texts. In phase 4, students agree on a joint position, which can be one of the positions in the texts or some compromise between the positions, and compose a joint, reasoned conclusion. The script allows some degrees of flexibility, i.e., students can use some time in a new phase to conclude the activities of the previous phase. Overall, the phases represent a progression from structured, narrowly focused activities to more free form, open activities.

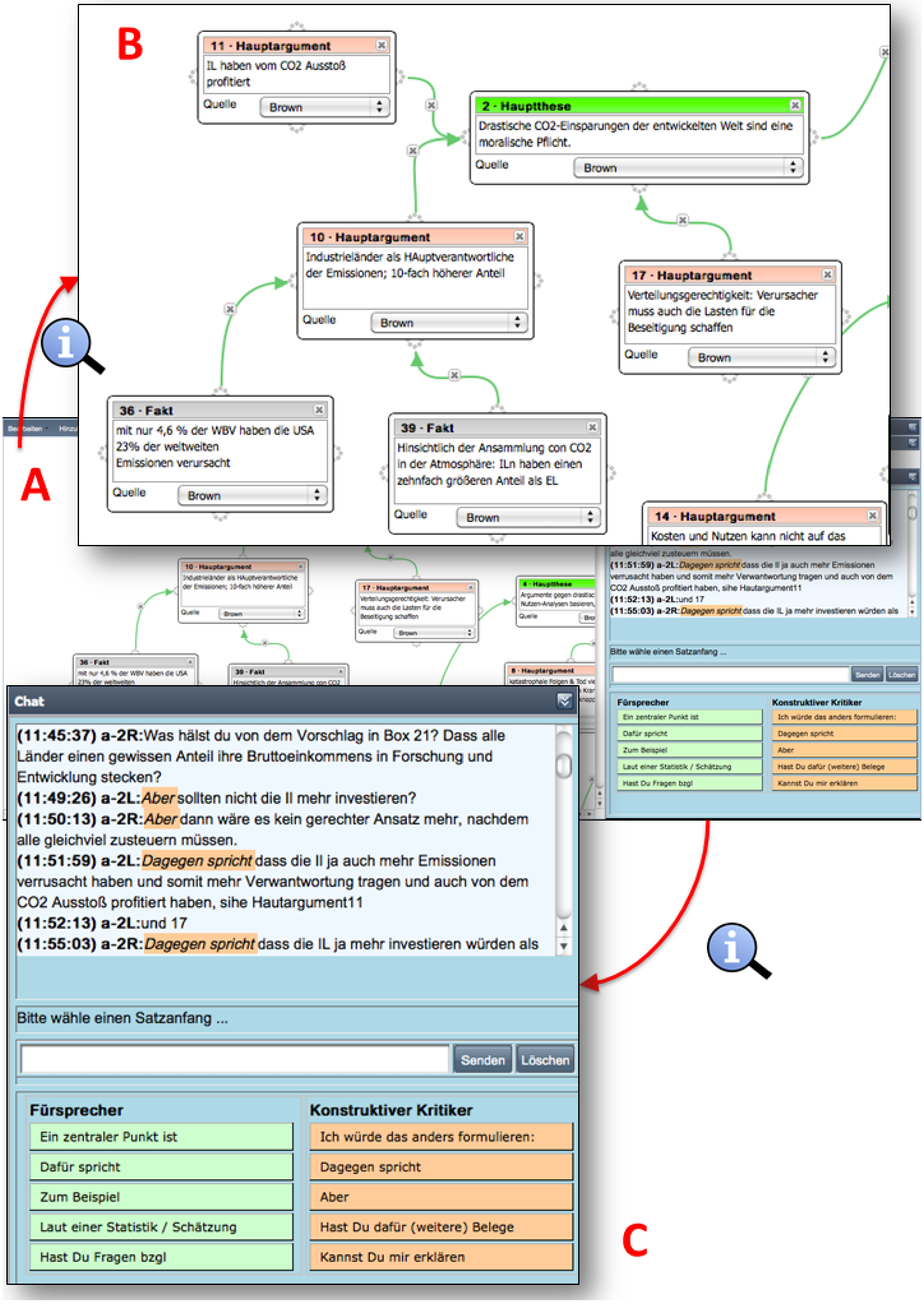
The approach has been termed *FACT-2* (*Fostering Argumentation through Conflicting Text, version 2*). A previous version of the approach, FACT-1, which shares some of the main principles but is based on simpler technologies (e.g., no argument diagrams and sentence openers), is described in Scheuer et al. (2011).

Figure 14 shows a screenshot of the overall LASAD setup used with FACT-2 (Figure 14.A), and detailed views on the diagramming tool (Figure 14.B) and the sentence opener interface (Figure 14.C). The buttons in the sentence opener interface allow students to choose from a predefined proponent and constructive critic moves to start the next chat message. The highlighted openings in the actual chat show previously used sentence openers.

Table 21
Sequence of activities used in the FACT-2 script

Phase	Individual / Collaborative	Description
1. Analyzing	individual	read assigned text model argument in a LASAD diagram
2. Discussing	collaborative	model argument in a LASAD diagram (cont'd) discuss individual texts
3. Interrelating	collaborative	discuss individual texts (cont'd) discuss relations (e.g., conflicts, agreements) between texts
4. Concluding	collaborative	agree on a joint position write down a justified joint conclusion

Note: Adapted from Scheuer et al. (2014).



11 - Hauptargument
IL haben vom CO2 Ausstoß profitiert
Quelle: Brown

2 - Hauptthese
Drastische CO2-Einsparungen der entwickelten Welt sind eine moralische Pflicht.
Quelle: Brown

10 - Hauptargument
Industrieländer als HAuptverantwortliche der Emissionen; 10-fach höherer Anteil
Quelle: Brown

17 - Hauptargument
Verteilungsgerechtigkeit: Verursacher muss auch die Lasten für die Beseitigung schaffen
Quelle: Brown

36 - Fakt
mit nur 4,6 % der WBV haben die USA 23% der weltweiten Emissionen verursacht
Quelle: Brown

39 - Fakt
Hinsichtlich der Ansammlung con CO2 in der Atmosphäre: ILn haben einen zehnfach größeren Anteil als EL
Quelle: Brown

14 - Hauptargument
Kosten und Nutzen kann nicht auf das

Chat

(11:45:37) a-2R: Was hälst du von dem Vorschlag in Box 21? Dass alle Länder einen gewissen Anteil ihre Bruttoeinkommens in Forschung und Entwicklung stecken?
(11:49:26) a-2L:*Aber* sollten nicht die II mehr investieren?
(11:50:13) a-2R:*Aber* dann wäre es kein gerechter Ansatz mehr, nachdem alle gleichviel zusteuern müssen.
(11:51:59) a-2L:*Dagegen spricht* dass die II ja auch mehr Emissionen verrusacht haben und somit mehr Verwantwortung tragen und auch von dem CO2 Ausstoß profitiert haben, sihe Hautargument11
(11:52:13) a-2L:und 17
(11:55:03) a-2R:*Dagegen spricht* dass die IL ja mehr investieren würden als

Bitte wähle einen Satzanfang ...

Fürsprecher	Konstruktiver Kritiker
Ein zentraler Punkt ist	Ich würde das anders formulieren:
Dafür spricht	Dagegen spricht
Zum Beispiel	Aber
Laut einer Statistik / Schätzung	Hast Du dafür (weitere) Belege
Hast Du Fragen bzgl	Kannst Du mir erklären

Figure 14: LASAD user interface configuration used for study: complete screen (A), detailed view on diagramming area (B), and detailed view on sentence opener interface (C). Adapted from Scheuer et al. (2014).

4.3 Study

Following from research question RQ1, the following hypothesis is formulated:

> **Hypothesis:** "A user interface that integrates argument diagramming with a discussion script (1) improves the quality of student-to-student interaction, and consequently, (2) leads to more student learning, compared to each method used individually."

To investigate this hypothesis, a quasi-experimental study was conducted at Saarland University, Germany, on July 8th – 9th, 2011, using a pretest-intervention-posttest design. The study compared two conditions:

- The *Script+* condition used the full version of the FACT-2 script, that is, the activity sequence in Table 21 including argument diagramming activities and the discussion script, as described above.
- The *Control* condition used an ablated version of the FACT-2 script. More specifically, the activity sequence in Table 21 including argument-diagramming activities was used as well but no discussion script. That is, students were not instructed to take on the roles of a proponent and a constructive critic, and the user interface did not include sentence openers but a standard chat tool instead.

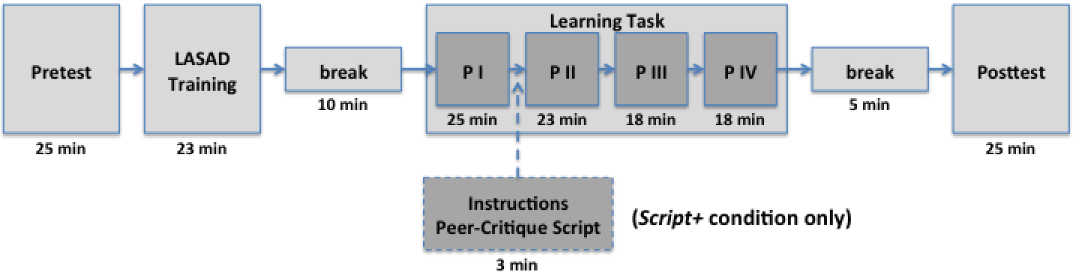


Figure 15: Sequence of activities during study. Adapted from Scheuer et al. (2014).

Note that, due to the limited number of available subjects, the study was set up in a way that only one aspect of the stated hypothesis could be statistically evaluated namely whether specifically the discussion scripting component makes a difference.

Forty-four (44) students enrolled in the Humanities and Social Sciences at Saarland University participated, yielding 12 Script+ dyads and 10 Control dyads. Figure 15 depicts the overall study procedure. The study procedure started with a pretest and a training task to familiarize students with the LASAD diagramming system. After a 10-minute break, students worked on the actual task as described in Table 21. Before the first collaborative phase (P II), the Script+ condition received short additional instructions regarding the peer-critique script (3 minutes). The actual task was followed by a 5-minute break before a posttest was administered.

Table 22

Coding scheme, its relation to the Rainbow framework (Baker et al. 2007), and example messages

Code	**Subsumed Rainbow codes**	**Example chat messages**
Low	1. Outside activity	"Just as a side note, how do you like this method?"
	2. Social Relation	"Hello" "How are you?" "Thank you."
	3. Interaction Management	"are you still writing?" "I cannot find box #57" "box #106, right hand side"
	4. Task Management (topic-unspecific)	"which main argument should we discuss first?" "let's start summarizing our conclusion" "we still need a justification"
Medium	4. Task Management (topic-specific)	"let's start with Lomborg's main thesis" "regarding helping argument #44:"
	5. Opinion	"what do you think [regarding our last point]" "could you give me an example for #15?" "whatever, I think Lomborg is right" "I agree" "then I misunderstood something"
High	6. Argumentation	"emission cuts are not only ethically ineffective but also factually unenforceable" "Lomborg's main thesis is well supported by #34 and #22" "Brown is right that ethical factors are not considered, #73"
	7. Broaden & Deepen	"Brown developed the main ethical idea well, Lomborg found a solution for it in #38" "well, actually, we should spend money on both issues [climate protection and support of poor countries]"

Note: Examples translated from German. Adapted from Scheuer et al. (2014).

To stimulate lively and critical discussions, climate change ethics was used as the discussion topic. Climate change is a complex and controversial problem involving uncertainty and ethical ramifications. As with many other real-world problems, there are multiple well-reasoned perspectives and no formally correct solution. The specific thesis to be discussed was *Developed countries have to cut their carbon emissions drastically*. Two argumentative texts with conflicting conclusions were used. Based on the recommendation of an expert in climate ethics teaching, two writers, one representing the *pro drastic emission cuts* position (Brown), the other the *con drastic emission cuts* position (Lomborg), were chosen. Two three-page summaries were composed based on selected writings of Brown and Lomborg. The text arguing against drastic emission cuts was based on Lomborg (2007). Lomborg argues for only moderate emission reductions based on a cost-benefit analysis. The main argument is that poor countries could be helped more effectively if money is spent otherwise (e.g., fighting hunger and diseases). The text arguing for drastic emission cuts was mainly based on Brown (2002). Brown argues for substantial emission reductions based on ethical obligations first-world countries have towards developing countries. The Brown summary was furthermore enriched with a discussion of several shortcomings of cost-benefit arguments in the global warming discussion taken from other Brown writings in order to better align the two opposing texts.

The following data was collected during the study (more details on the specific instruments and analysis approach can be found in Scheuer et al. [2014]):

- *Relevant student characteristics* were elicited through multi-choice questions (MCQs) in the pretest to determine the homogeneity of conditions. These characteristics include: age, gender, course of studies, interest in the topic climate change, and attitude towards collaborative learning, computer-based chat, visually represented information, and argumentation.
- *Discussion quality* was evaluated based on an analysis of chat protocols. The protocols were segmented into sentence-level units based on punctuation marks. Using the Rainbow framework as a guide (Baker et al. 2007), a coding handbook was developed, distinguishing three levels of elaboration (*Low*, *Medium*, and *High*). Table 22 shows the coding scheme. The coding manual was validated in terms of the inter-rater reliability between two independent coders using Cohen's kappa (1960) with a satisfactory result (κ = .76). Based on the coding handbook, each protocol segment was assigned one of the

elaboration codes. For each chat protocol, the number of codes per category was aggregated into an overall score ("code-and-count" approach).

- *Subjective learning gai*ns were elicited through MCQs in the posttest, in particular, perceived learning of domain knowledge (*climate ethics*), argumentation theory (*knowing about* argumentation), and argumentation practice (*doing* argumentation).
- *Objective knowledge of the two texts* was measured through MCQs, focusing on detailed factual knowledge.

4.4 Results

Before reporting the quantitative results, a concrete individual case is discussed to give the reader a sense of the learning task and how students tackled it.

4.4.1 Case analysis

Table 23 presents an excerpt from a Script+ chat. Text in italic typeface indicates the use of sentence openers. In the original user interface, sentence openers are highlighted green (for proponent contributions) and red (for critic contributions). Student S1 read the Brown text (pro drastic emission cuts) and student S2 the Lomborg text (anti drastic emission cuts). The selected case demonstrates one extreme—a discussion segment in which sentence openers have been consequently used. The excerpt illustrates how the peer-critique script can influence student discussions in both positive and negative ways.

Despite the option of composing free-text messages, the dyad makes consistent use of the provided structuring, using a sentence opener in every message without exception. Also, they make appropriate use sentence openers, perhaps with the exception of turn #3, in which the sentence opener indicates the modification of a previous statement while the message body is a straight counterargument against a prior statement. In many cases there is evidence of good argumentation behavior. For instance, student 1 substantiates his claim in turn #1 through the data provided in turn #2. Student 2 justifies his claim in turn #3 through the reason given in turn #5, and illustrates this reason through the examples given in turn #7. Students also adopt an informed and academic style of arguing by citing numbers and statistics (turn #2 and #9). Possibly, this behavior was triggered through the explicit representation of *fact* elements in the argument diagram. Overall, in line with the quantitative analysis below, the discussion is very task-focused without any off-topic talk. Many arguments from the texts are cited and connected to one another.

Table 23

Excerpt from a Script+ condition chat

#	St.	Contribution
1	S1	*A central point is* that climate change can only be controlled through drastic emission cuts
2	…	*An argument for this point is* that according to an estimate of the IPCC global emissions must be cut by 60% to keep the amount of greenhouse gases in the atmosphere at a safe level
3	S2	*I'd phrase this differently:* Investments into climate protection can cause similar harm.
4	S1	*Do you have (any / more) evidence*
5	S2	*A central point is* that we might forget about other important social projects
6	S1	*A central point is* that industrial countries must accept the main responsibility to solve the climate change problem and to reduce their use of the atmosphere
7	S2	*For instance* there won't be any money for schools, hospitals, etc.
8	S1	*An argument for this point is* that industrial countries are responsible for the main share of emissions
9	…	*For instance* the USA is responsible for 23 percent
10	…	*A central point is* that cost-benefit analyses are based on ethically dubious arguments
11	S2	*Do you have (any / more) evidence*
12	S1	*An argument for this point is* that they are biased (e.g., oil industry)
13	…	*A central point is* that the quantification of costs and benefits of climate protection measures is a general problem

Note: Translated from German. Sentence openers set in italic type. Triple-dots in the second column indicate that the current speaker continues his or her turn. Adapted from Scheuer et al. (2014).

On the downside, the students' discussion has more breadth than depth, i.e., points are mentioned, briefly explained, but not critically elaborated. In turn #3, student 2 objects to his partner's conclusion. From then on, both students are solely focused on supporting their own position without referring to what their partner has said (except for some generic clarification questions, turn #4 and #11). A closer look at the selected sentence openers reveals that, apart from turn #3, only proponent moves are used, which is in accordance with the observation that critical references to the partner's contributions are missing. Also, the arguments stem from copying from the argument analysis with little co-elaboration of new meanings and ideas. On the positive side, this clearly shows that the argument diagram has been used as a resource for structured discussions. Another observation is that, due to the consequent use of sentence openers, this discussion has a somewhat stilted feel, pointing at the possible danger that the script suppresses natural interactions

(Dillenbourg 2002; Fischer et al. 2013). A possible countermeasure may be to replace sentence openers with message labels (e.g., *claim*, *pro-argument*, *data*, *objection*, *clarification request*) and leave the decision which specific formulations to use to the student. This would prevent the repetitive nature of formulations and, at the same time, increase freedom and agency of students. In sum, while promoting a well-reasoned dialogue, the script was not very successful in engaging students in critical interactions with one another. The question whether this tendency is representative of the whole sample will be addressed when discussing RQ1 below.

4.4.2 Homogeneity of conditions and general population characteristics

The analysis of student characteristics (age, gender, course of studies, relevant interests and preferences) showed no significant differences between the Script+ and Control conditions. Therefore, possible influences of these characteristics on study results are assumed negligible. In general, participants were on average 22.7 years old (*SD* = 3.0) and in their 5th semester (*M* = 4.9; *SD* = 3.2). The majority of participants were female (64%). 45% of the participants studied humanities (e.g., philosophy, languages, history, and cultural sciences). Another 45% were enrolled in pre-service teacher education programs. The remaining 10% studied psychology.

4.4.3 Use of sentence openers

On average, Script+ students used a sentence opener (SO) in chat messages in one out of five messages (20%), that is, SOs were used considerably more as in the Lazonder et al. (2003) study. The use of SOs differed notably between dyads (see Figure 16, left): five dyads made frequent use (> 25% SO messages), three dyads made occasional use (> 10% SO messages), and four dyads made rare use of SOs (< 10% SO messages). Note that 25% SO messages is already quite considerable since a substantial portion of each dialogue is about greeting, interaction and task management, off-topic talk, etc., rather than about subject matters (36% *Low* elaboration moves, see Table 24). Critic SOs (54%) were used slightly more often than proponent SOs (46%; see Figure 16, right). There are considerable differences in the use of specific SOs: Three SOs were frequently used (> 15% of all SO uses), two SOs were occasionally used (> 10% of all SO uses), and five SOs were rarely used (< 10% of all SO uses). An analysis of intra-class correlations (ICC) revealed that students within the same dyads significantly influenced each other in terms of the extent of SO uses (*ICC* = 0.83; *F* = 10.6; *p* < .001). That is, if one student decided to use (or not to use) SOs, it was very likely that the other student decided so as well.

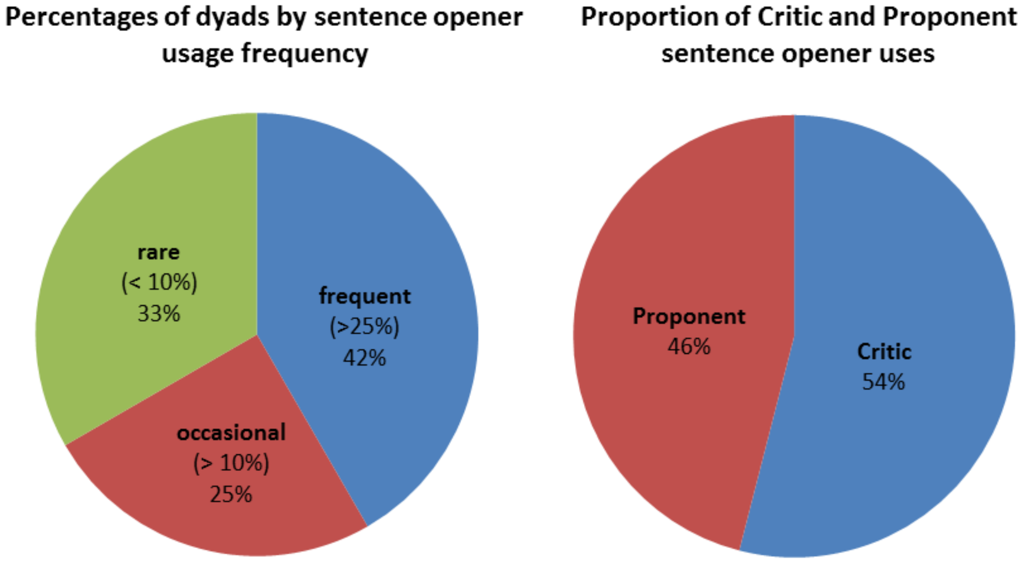


Figure 16: Frequency of sentence opener use. Left chart: Percentage of dyads who used sentence openers frequently (i.e., sentence openers used in more than 25% of all messages), occasionally (SOs in between 10% and 25% of all messages), and rarely (SOs in less than 10% of all messages). Right chart: Proportion of Critic versus Proponent sentence opener uses.

4.4.4 Discussion quality

Table 24 shows the results of the chat analysis, both, in terms of the average absolute number of codes (see also Figure 17) and the average proportion of codes per chat (see also Figure 18). On the one hand, absolute numbers have the disadvantage that they do not allow controlling for chat length. That is, participants with a lower posting frequency have fewer opportunities for producing chat messages of each kind, resulting in lower scores on each dimension. On the other hand, proportional values must also be taken with a grain of salt. For instance, when analyzing the amount of *High* codes, proportional values penalize chats with many *Low* or *Medium* contributions, which are not per se indicative of bad collaborative behavior.

The results show that Script+ chats were on average shorter, with less *Low* and *Medium* elaboration messages and more *High* elaboration messages, both in terms of absolute and relative numbers. Maybe the most interesting result is a significant difference in terms of *High* codes in favor of the Script+ condition, a result in accordance with the hypothesis. That is, students in the Script+ condition were proportionally more engaged in the intended behaviors. In terms of the absolute number of *High* codes, there is a non-significant trend (p = .07) with large effect size (d = 0.82) favoring the Script+ condition. The results of the *Low* category show the opposite pattern. The proportion of *Low* codes in the control condition was significantly higher compared to the Script+ condition. A possible interpretation is that Script+ dyads were more task-focused and therefore used less *Low* messages, which do not or only marginally contribute to the discussion goal. This interpretation is in line with previous results indicating that discussion scripts help learners keep a focus on the learning task (Baker and Lund 1997; Oh and Jonassen 2007). Again, in terms of absolute numbers, the difference indicates a strong effect (*d* = -0.82), yet, does not reach a significant level (*p* = .06).

Table 24
Statistical summary of chat analysis (Script+ vs. Control)

	Script+		**Control**		**Differences**			
	M	***SD***	***M***	***SD***	***abs***	***t***	***p***	***d***
# Total	83.2	29.8	94.8	36.6	-11.6	0.82	.42	-0.35
# Low	31.1	17.3	50.6	28.6	-19.5	1.96	.06	-0.82
# Medium	13.0	4.6	18.6	11	-5.6	1.61	.12	-0.66
# High	38.9	17.9	25.6	14.4	13.3	1.90	.07	0.82
% Low	36	11	54	23	-18	2.36	.03	-0.98
% Medium	16	5	18	10	-2	0.68	.50	-0.29
% High	48	11	28	16	20	3.52	.00	1.48

Note: Adapted from Scheuer et al. (2014).

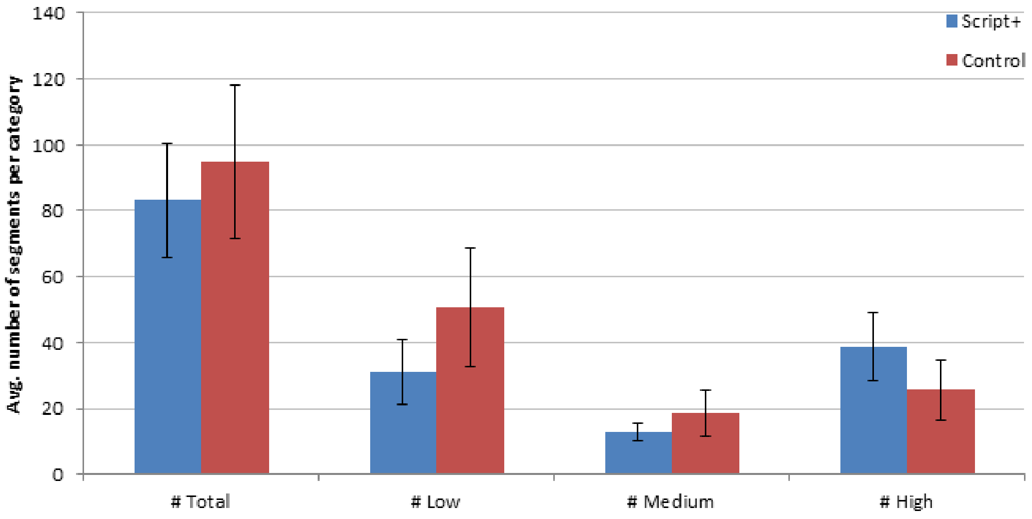


Figure 17: Average number of discussion segments coded as *Low*, *Medium*, and *High* elaboration move. Error bars indicate 95% confidence intervals.

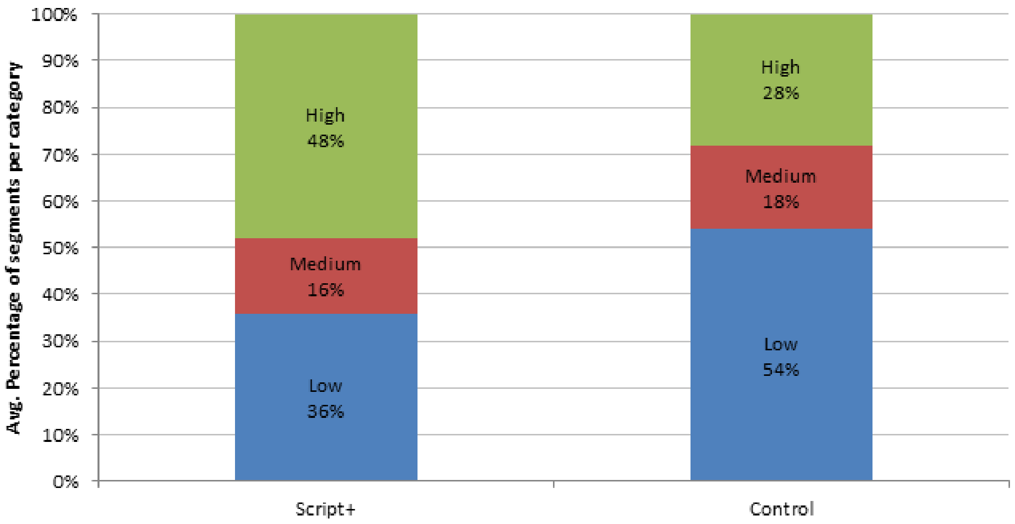


Figure 18: Average percentage of discussion segments coded as *Low*, *Medium*, and *High* elaboration move.

4.4.5 Subjective learning

The research hypothesis is further supported by the post-test analysis, which indicates that Script+ students assessed their learning more positively than Control students (see Figure 19). In terms of argumentation practice, there was a significant difference favoring the Script+ condition (*p* = .01; *d* = 0.88). In terms of argumentation theory, there was a non-significant trend (*p* = .07; *d* = 0.58). In terms of domain knowledge, there was no difference (*p* = .26; *d* = 0.35).

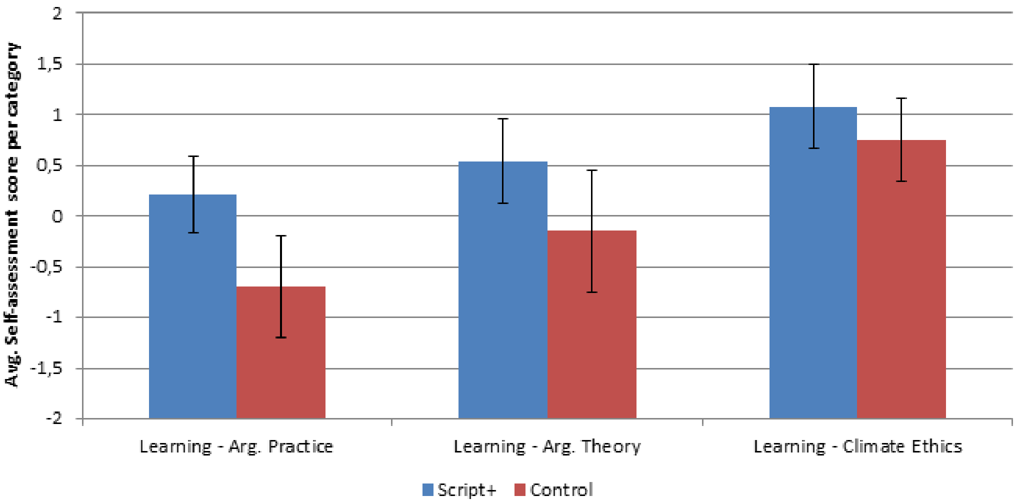


Figure 19: Average student self-assessments of their learning regarding argumentation practice, argumentation theory, and climate ethics. Error bars indicate 95% confidence intervals.

4.4.6 Objective learning

On average, students answered slightly more than half of all MCQ questions correctly (*M* = 54; *SD* = 20). There was no difference in means in terms of Brown and Lomborg questions (*M* = 54, *SD* = 28 [Brown], *SD* = 32 [Lomborg]).

Table 25
Statistical summary of posttest MCQ analysis (Brown vs. Lomborg)

	Brown items		**Lomborg items**		**Differences**			
	M	***SD***	***M***	***SD***	***abs***	***t***	***p***	***d***
Brown students	72	20	38	24	34	5.98	.00	1.55
Lomborg students	36	24	70	31	-34	5.50	.00	-1.23

Note: Adapted from Scheuer et al. (2014).

Table 25 shows a statistical comparison between the Brown and Lomborg item scores for students who were assigned the Brown text (first row) and students who were assigned the Lomborg text (second row). Not surprisingly, students performed significantly better on items that elicited information from "their" text (in both cases: *p* < .001). For MCQs that tested students on the text they modeled in LASAD, they scored, on average, 72 points (for Brown students; *SD* = 20) and 70 points (for Lomborg students; *SD* = 31). For the text they only knew from a partner's diagram and discussion with that partner, students scored, on average, 38 points (Brown students on Lomborg questions; *SD* = 24) and 36 points (Lomborg students on Brown questions; *SD* = 24). The cross-text scores are well above the chance rate of 25 points but clearly lower than the score obtained for "their" own text. One reason why students did not do better might be that very specific and detailed information was elicited in the questions (e.g., whether an approach proposed by Lomborg would save more or about the same amount of human lives compared to measures in accordance with the Kyoto protocol). Students might remember such information from a careful reading of the text, but it is less likely they would have discussed that level of detail in their chat discussions. The hypothesis that the discussions contributed little to the achieved MCQ scores is further supported by the fact that the number of *High* elaboration moves did not correlate with the average MCQ score per dyad (*r* = .08; *t* = 0.36; *p* = .73). Research on knowledge maps (a variation of concept maps [Novak 1990]) suggests that the reading of diagrammatic representations has positive effects on the memory of main ideas rather than on the memory of subordinate ideas (O'Donnell et al. 2002). This could explain why exposure to a partner's diagram did not have a major influence on the MCQ posttest scores for students, since the MCQ test questions were focused more on subordinate facts than the main ideas of the texts.

Table 26 and Figure 20 show a statistical comparison between Script+ and Control students with respect to MCQ scores. In total, and on both sub-scales, there were no differences between the Script+ and the Control condition. This is in accordance with the assumption above: Students' knowledge of text details is mainly based on the modeling task rather than the discussion task, so the treatment, which only took effect during the discussions, apparently had marginal, if any, influence on this aspect of students' knowledge gains.

Table 26

Statistical summary of posttest MCQ analysis (Script+ vs. Control)

	Script+		**Control**		**Differences**			
	M	***SD***	***M***	***SD***	***Abs***	***t***	***p***	***d***
Total	53	22	55	19	-2	0.28	.78	-0.09
Brown items	53	30	55	27	-2	0.19	.85	-0.06
Lomborg items	53	32	55	32	-2	0.19	.85	-0.06

Note: Adapted from Scheuer et al. (2014).

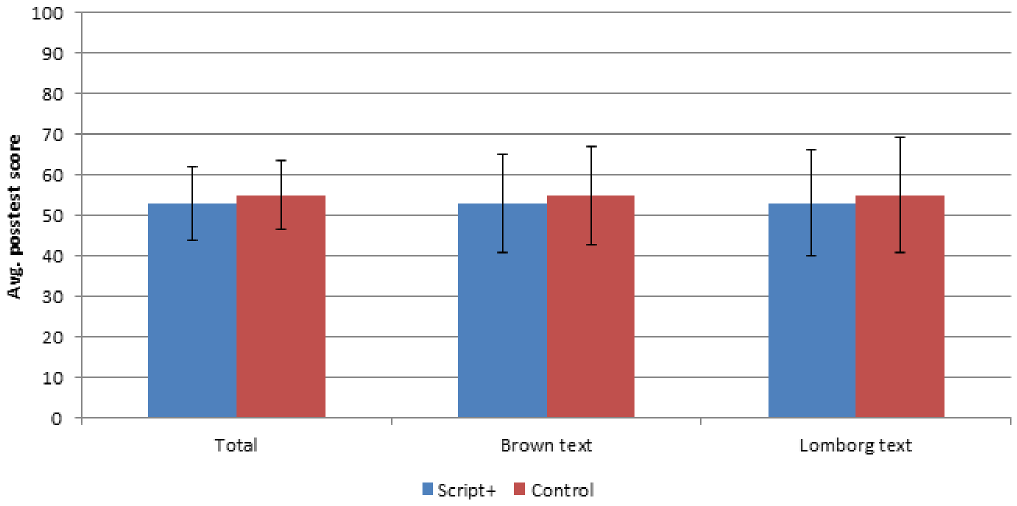


Figure 20: Average posttest score regarding overall knowledge about the texts, the Brown text, and the Lomborg text. Error bars indicate 95% confidence intervals.

4.5 Discussion and Conclusion

Table 27 summarizes the results achieved with FACT-2. Overall, there is evidence in favor of the researched hypothesis both in the chat protocols and in the participants' assessment of their learning: A discussion script, used in combination with an argument diagram, can promote the quality of student discussions. Benefits have been found on different dimensions: a significant effect with large effect size regarding the proportion of elaborative moves (i.e., arguments, counterarguments, and explanations; Table 27, second row), a significant effect regarding students' assessment about whether they learned to argue better (Table 27, third row), and a trend regarding students' assessment whether they learned about argumentation (Table 27, third row). A closer look at the Script+ condition (i.e., the full version of FACT-2) showed that students indeed utilized the sentence openers to their advantage: Two thirds of Script+ dyads made frequent or occasional use of sentence openers (Table 27, first row); others may have oriented themselves by reading the given sentence openers while using their own formulations. There was a substantially increased use of sentence openers compared to the Lazonder et al. (2003) study. A possible explanation is that the diagrams prompted and guided students in selecting appropriate sentence openers. Anecdotal evidence shows that students reused text fragments from their argument analysis and combined them with appropriate sentence openers to compose chat messages. Thus, combining argument diagrams and sentence openers might indeed have led to benefits that could not have been achieved with either method alone.

No differences were found in terms of detailed knowledge of the given texts (Table 27, last row). As expected, students had much better knowledge of the text they read and then modeled themselves as compared to the text they learned about from their partner's modeling and subsequent discussion between the partners. Yet, the approach of reading and modeling one text, while reviewing an opposing text through discussion with a partner, may have been beneficial. Rather than focusing on subtle details of one text in isolation, which was tested for in the posttest, the setup of this experiment appeared to prompt students to interrelate and critically evaluate positions and arguments, which may have led to a broader, multi-perspective understanding of the topic at hand and the emergence of new perspectives (Wegerif et al. 2010). Previous results show that engagement in dialectical argumentation indeed promotes deeper processing and conceptual change (Asterhan and Schwarz 2009). The presented analysis does not allow for a definitive conclusion on this, which is in general hard to assess due to a wide variety of sometimes surprising and unpredictable insights students gain from fruitful learning discussions.

Table 27
Summary of results

Analyzed aspect	**Result**
Use of sentence openers	two-thirds of all Script+ dyads made frequent or occasional use of sentence openers *critic* sentence openers were used slightly more often than "proponent" sentence openers (54% vs. 46%)
Discussion quality	significant effect in terms of the proportion of High elaboration (in favor of the Script+ condition) (non-significant trend with large effect size when comparing absolute numbers of codes rather than proportions)
Subjective learning	significant effect in terms of perceived learning of argumentation practices (in favor of the Script+ condition) (non-significant) trend in terms of perceived learning of argumentation generally (in favor of the Script+ condition) no difference in terms of perceived learning about the topic of climate ethics
Objective learning	no difference in terms of detailed knowledge of the given texts

Note: Adapted from Scheuer et al. (2014).

There is good reason to believe that the FACT-2 script and the design principles it is based upon can also be successfully applied in domains other than climate ethics. As long as there are two (or more) opposing positions that can be represented in a diagram, FACT-2 can be employed to guide the analysis and discussion of these positions. For instance, one could imagine using FACT-2 to incite discussions about other socio-ethical dilemmas, which naturally have different reasonable perspectives, such as the use of genetically modified food (Munneke et al. 2003), experiments on animals (McLaren et al. 2010), or the right to die (Cavalier and Weber 2002). Similarly, FACT-2 can be applied for planning and design problems, which often require the reconciliation of opposing design concerns. For instance, engineering students could investigate the tradeoffs between technical efficiency, financial costs, and environmental friendliness in the design of technical systems (Chaudhuri et al. 2009). Instructional science students could review and discuss the controversy on the usefulness of collaboration scripts as an instructional instrument as opposed to more free-style forms of collaboration (Dillenbourg 2002). Other possible application areas are manifold, including politics, history, science, medical decision-making, and the Law. Depending on the specific domain and pedagogical objectives, certain adjustments may be needed, for instance, different categories in the diagram (e.g., *hypothesis* elements in the science domain), different sentence openers (e.g., the opener *According to a statistic / estimate* ... may or may not be appropriate for legal debates), or different time allocations for the specific phases may be used.

In conclusion, the presented results are in line with previous research underscoring that the quality of collaboration / argumentation practices can be improved through appropriate structural support (e.g., McAlister et al. 2004; Rummel and Spada 2005; Weinberger et al. 2010). In particular, peer-critique scripts (e.g., Andriessen et al. 2003; Clark et al. 2009; Johnson and Johnson 1994; Weinberger et al. 2005) may help students to engage in positive collaborative behaviors that would otherwise not (or only to a lesser extent) occur.

The investigated setup differs from previous discussion scripting approaches in several respects. In many previous studies, topics within the psychology and education curriculum have been addressed (e.g., Schellens et al. 2007; Stegmann et al. 2007, 2012; Weinberger et al. 2005; Weinberger et al. 2010). In contrast, climate ethics was chosen for the study reported here. Moreover, the used setup is unique by employing a combination of argument diagrams and sentence openers. Overall, by covering a discussion domain and setup not researched much, or at all, the present study broadens the empirical basis of CSCL script theories.

Closest to the presented work are probably previous designs that combine social scaffolding with some structuring at another level. The approach by Rewey et al. (1989, 1992) superficially seems to be similar to FACT-2, since it also combines a graph-based representation of knowledge with a collaboration script. Yet, a closer look reveals a number of notable differences. First, their knowledge maps represented domain knowledge (more specifically, medical procedures like taking blood pressure), rather than lines of argumentation. Second, their social script only consisted of instructions regarding the main activities to engage in and did not include sentence openers, which is the main component of the social script employed within FACT-2. Hence, social interaction was scripted on a much more coarse level, roughly comparable to the four phases of the macro-script of FACT-2. Third, the kind of activity promoted by the script was different in that reciprocal summarization activities were aimed at, not argumentation. Finally, Rewey et al. (1989, 1992) did not investigate the effect on student discussions or other kinds of interactions, but rather focused on the acquisition of the domain knowledge represented in the knowledge maps or in alternative formats. Another in some respect similar approach is described in Weinberger et al. (2005; first study). The learning task was to collaboratively analyze given cases in terms of a specific theoretical framework in social psychology concerned with attributions—the subject matter to be learned. They used social scaffolding in form of sentence openers, which were part of a role script, similar to the sentence openers of FACT-2. As epistemic scaffolds they used prompting questions, which were displayed when composing the first message to a discussion thread. The prompting questions of the epistemic script aimed at providing guidance to the analysis process. For instance, one of the prompts was “Is the cause for the attribution stable or variable?” That is, while their social script is in some respect similar, their epistemic script is considerable different to the diagrams used in FACT-2.

Four issues arise from the present study as possible avenues for future research. First, a main motivation of the used setup was to achieve synergistic scaffolding effects between argument diagrams and sentence openers. Anecdotal evidence hinted at possible interactions between diagrams and sentence opener use. Yet, because a one-factorial design was employed, which only included the peer critique script as an independent variable, it was not possible to statistically quantify the effect diagrams had—neither a possible main effect, nor a possible interaction effect with the peer critique script could be tested for. Therefore, future studies may employ 2×2 design to precisely determine what contribution the diagrams made in improving the learning process, and whether they interacted with the peer critique script. Such an interaction would give a strong and direct indication of the hypothesized synergistic effect.

Second, the present study could not confirm that the peer-critique script had a positive impact on the acquisition of domain-specific knowledge and reasoning skills. A possible reason is that the posttest knowledge test, which focused on factual knowledge questions, was not appropriate to capture the kind of deep conceptual learning, which is assumed to be promoted by argumentative activities (Asterhan and Schwarz 2009). Future research might address this issue more explicitly, for instance, based on an analysis of an essay-writing task conducted before and after the intervention (e.g., “Make a statement as discerning and well-reasoned as possible concerning the question ‘...’ under inclusion of both, supporting and opposing arguments.”). Such an analysis could reveal whether students (1) have changed their perspective on the given topic based on evidence and reasoned argument, (2) can provide more substantial support for their perspective, (3) are more willing and capable to acknowledge arguments and facts favoring positions that oppose their own, (4) are aware of different value systems underlying different perspectives, and (5) know about arguments that they did not know before or even ones that were not included in the background readings. A similar question could target a new topic in order to test whether a general attitudinal change towards other perspectives occurred, that is, whether students are more able to see the pros and cons of the different positions, independent of their own initial positions, and, based on this, conceive an informed opinion, which is potentially synthesized from elements of opposing standpoints. This would indicate a step towards an evaluative epistemological stance, an attitude that is positively linked to argumentative skill development (cf. Kuhn 1991, pp. 172–203).

Third, another question that calls for further investigation is whether students maintain improved argumentation practices when no structural support is available (cf. Pea 2004). This could be tested in a post-intervention application phase (similar to the experimental setup used by Rummel and Spada [2005]), in which students engage in collaborative argumentation using a discussion environment without sentence openers. The resulting chat traces could be analyzed using the same coding approach that was used to code the intervention chats. In addition, it could be checked whether Script+ students make increased use of the sentence openers they had available to them during the intervention.

Fourth, the provided structuring may not only guide students to engage in higher-quality argumentation. In addition, the resultant structuring of user input can also potentially be exploited to facilitate an automated analysis of interactions. This issue will be discussed in greater detail in Chapter 9.

Chapter 5
Designing a Configurable Argumentation Support Engine

Similar to the approach discussed in the previous chapter, most prior approaches to supporting argumentation are static in nature (e.g., Stegmann et al. 2007; Suthers and Hundhausen 2003). That is, the provided structuring does not adapt to the situational requirements that emerge during the process. While static approaches can provide structure and guidance to help students engage in desired forms of argumentation, they are neither able to recognize and remediate when students struggle, nor can they adjust to the specific needs of individual students. It would be desirable to imbue systems with the ability to automatically analyze the behavior, progress, problems, and learning of students, and to tailor support accordingly. Such adaptive support may include feedback regarding the current problem solving, e.g., potential errors in the current version of a solution, or hints what can be done next if students reach an impasse. Another form of adaptation is to dynamically reduce the amount of structuring and support as students become more competent. For instance, Wecker and Fischer (2011) used a relatively simple approach of *fading the scaffold* to support online collaboration. In their approach, the amount of instructional support presented in the user interface was reduced based on the number of contributions students already made. Developing and researching computational adaptation mechanisms has been the province of the intelligent tutoring systems community for several decades now (VanLehn 2006; Woolf 2008). In recent times, research interests expanded from well-structured domains, such as algebra problem solving, to more ill-defined domains, including argumentation and collaborative learning (Magnisalis et al. 2011; Soller et al. 2005).

Despite considerable advances made in research, the penetration of intelligent and adaptive educational technologies in real-world educational contexts is still relatively limited. One of the main barriers hindering a wider adoption is the high cost incurred in developing and customizing intelligent and adaptive technologies. The research field of intelligent tutoring authoring systems addresses this problem, aiming at tools that allow building adaptive, interactive, and intelligent educational software in a cost-effective manner (Murray et al. 2003). The work described in this chapter contributes to this field of research.

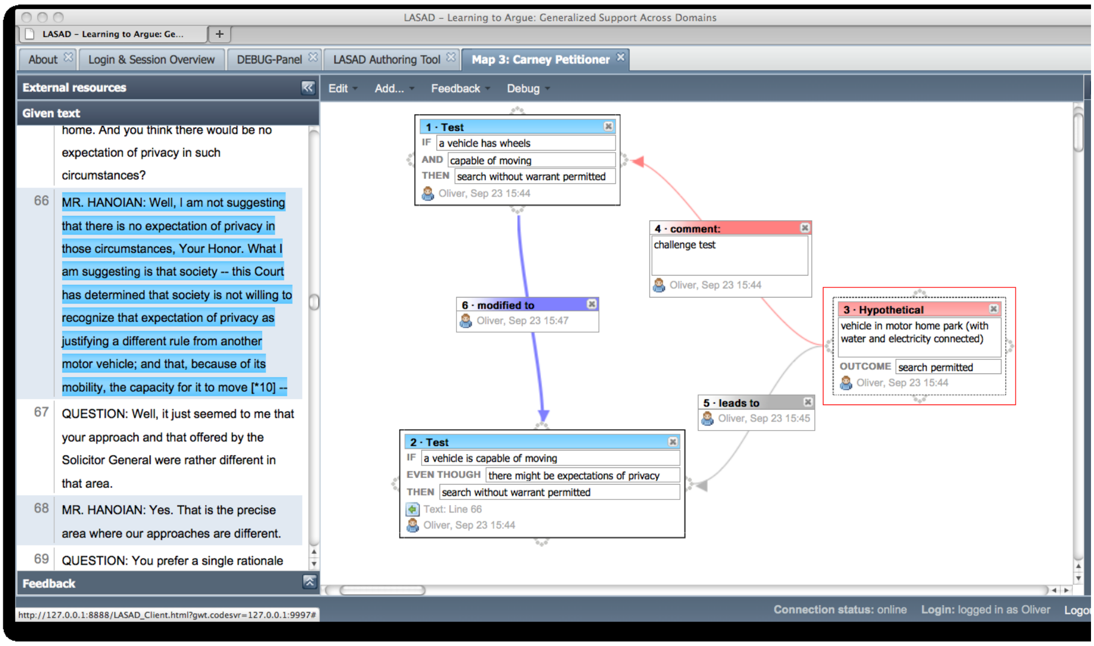
This chapter presents the *CASE* engine (*Configurable Argumentation Support Engine*), which allows the flexible definition of automated, adapted support across different argumentation domains and learning scenarios. The design of the CASE engine has been informed through an extensive analysis and systemization of past approaches, discussed in section 3.4 and section 3.5. The CASE engine formalizes crucial dimensions of the design space of adaptive argumentation learning support, generalizing across approaches successfully used in past systems.

Section 5.1 introduces the LASAD argumentation framework, which the CASE engine is part of. Section 5.2 motivates and formulates the specific research questions addressed with the CASE engine. The following sections focus on the technical design of the CASE engine, in particular, the overall concept (section 5.3), the software architecture (section 5.4), knowledge representations and inference mechanisms within the CASE engine (section 5.5), and the specific configuration options available (section 5.6). Section 5.7 discusses four showcase applications, which demonstrate the breadth of applicability of the CASE approach—the main design objective pursued. Section 5.8 presents a graphical user interface to configure and manage adaptive support functionalities—the most recent development in context of the CASE engine—and reports on an empirical study on the usability of the tool. Finally, section 5.9 discusses and summarizes the results of this chapter.

5.1 Background: The LASAD Project

The LASAD project was motivated by the observation that argumentation learning systems can typically not be easily adapted to new requirements, since they tend to be tied to specific argumentation domains, visualizations, or collaboration modes. Yet, on a conceptual level, these systems share many features in terms of the user interface and underlying functionality. In principle, it should be possible to develop a more general framework that can be used as a foundation for building specific argumentation systems in a more simplified fashion, based on well-defined configuration and extension mechanisms. Within the LASAD project, this was precisely the objective and what has been developed.

At its heart, the LASAD system (Loll et al. 2012) supports individuals or groups of learners to create argument diagrams in (shared) workspaces. As a general, cross-domain framework, LASAD enables users (i.e., developers, teachers, and researchers) to configure workspaces according to their specific requirements. Communication and task-related tools can be added to the workspace, such as a text chat, a sentence opener interface, and a text display that allows linking of text passages to elements of the argument diagram. Boxes and links can be configured differently per application; labels, visual appearance, and subcomponents (e.g., text fields, radio buttons and dropdown menus) can be altered. LASAD is purely web-based; a modern web-browser and web access is all that is required to use the system. The generality of LASAD has been shown through its use in a wide variety of differently targeted argumentation-learning applications and empirical studies. Besides original uses of LASAD, for instance, within the LASAD project (Loll and Pinkwart 2013; Scheuer et al. 2014), the Metafora project (Dragon et al. 2013), and the ArgumentPeer project (Lynch and Ashley 2012), a number of well-known existing argumentation systems have been emulated with the LASAD framework, including Belvedere (Suthers et al. 2001; see Figure 7, p. 103 [original] and Figure 21 [emulated]) and LARGO (Pinkwart et al. 2009; see Figure 22 [emulated]).



home. And you think there would be no expectation of privacy in such circumstances?

66 MR. HANOIAN: Well, I am not suggesting that there is no expectation of privacy in those circumstances, Your Honor. What I am suggesting is that society -- this Court has determined that society is not willing to recognize that expectation of privacy as justifying a different rule from another motor vehicle; and that, because of its mobility, the capacity for it to move [*10] --

67 QUESTION: Well, it just seemed to me that your approach and that offered by the Solicitor General were rather different in that area.

68 MR. HANOIAN: Yes. That is the precise area where our approaches are different.

69 QUESTION: You prefer a single rationale

1 · Test
IF a vehicle has wheels
AND capable of moving
THEN search without warrant permitted
Oliver, Sep 23 15:44

4 · comment:
challenge test
Oliver, Sep 23 15:44

6 · modified to
Oliver, Sep 23 15:47

3 · Hypothetical
vehicle in motor home park (with water and electricity connected)
OUTCOME search permitted
Oliver, Sep 23 15:44

5 · leads to
Oliver, Sep 23 15:45

2 · Test
IF a vehicle is capable of moving
EVEN THOUGH there might be expectations of privacy
THEN search without warrant permitted
Text: Line 66
Oliver, Sep 23 15:44

http://127.0.0.1:8888/LASAD_Client.html?gwt.codesvr=127.0.0.1:9997#

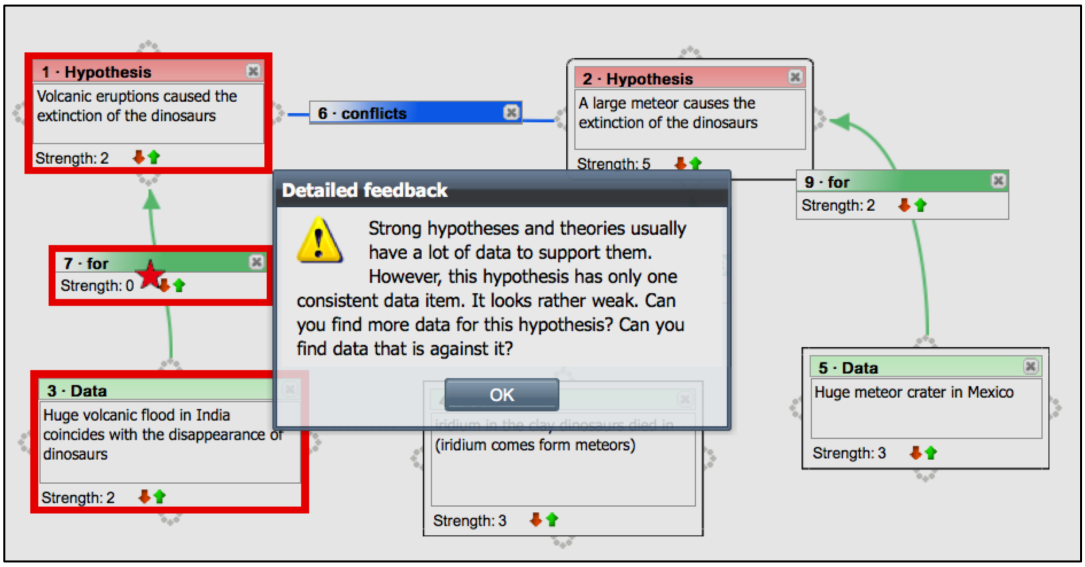
Connection status: online Login: logged in as Oliver Logo

Figure 21: LASAD user interface, configured to emulate the LARGO system (Pinkwart et al. 2009).

Figure 21 shows a LASAD configuration to emulate the LARGO system for legal argumentation (Pinkwart et al. 2009), illustrating some basic features of LASAD. The panel on the left contains a legal transcript (a protocol of the argumentative exchange between lawyers before the US Supreme Court) that students analyze by creating an argument diagram in the panel on the right. Students can create different types of boxes (e.g., *Hypothetical*, *Test*) and links (e.g., *leads to*, *modified to*).

Different sub-elements are available for the different box and link types. For instance, *Hypothetical* boxes consist of an unlabeled text field and an *Outcome* text field, while *Test* boxes allow defining logical rules by filling in text fields for a condition (*IF* text field) and a conclusion (*THEN* text field). Through a menu accessible in the title bar of the box, students can extend the condition by dynamically adding new condition clauses (*AND* text fields). Boxes and links are configured to display awareness information, such as the creator and the creation time (see the line at bottom of boxes).

Figure 22 shows a LASAD configuration that emulates the Belvedere system for scientific reasoning (Suthers et al. 2001), illustrating the adaptive feedback within the LASAD system. The feedback message in the window on top of the LASAD workspace prompts students to search for additional data to evaluate a specific hypothesis. The message refers to the diagram elements highlighted in red, namely, a hypothesis (box number 1) with only one supporting piece of data (box number 3). Students can request such hints from LASAD by selecting a *Get hint* entry from the feedback menu (see the menu bar in Figure 21, p. 157). The configuration and generation of feedback messages is done within the CASE engine, which has been developed as part of this dissertation and will be the focus of this chapter.



1 · Hypothesis
Volcanic eruptions caused the extinction of the dinosaurs
Strength: 2

6 · conflicts

2 · Hypothesis
A large meteor causes the extinction of the dinosaurs
Strength: 5

9 · for
Strength: 2

7 · for
Strength: 0

Detailed feedback

Strong hypotheses and theories usually have a lot of data to support them. However, this hypothesis has only one consistent data item. It looks rather weak. Can you find more data for this hypothesis? Can you find data that is against it?

3 · Data
Huge volcanic flood in India coincides with the disappearance of dinosaurs
Strength: 2

(iridium comes form meteors)
Strength: 3

5 · Data
Huge meteor crater in Mexico
Strength: 3

Figure 22: Detail of LASAD user interface, configured to emulate the Belvedere system (Suthers et al. 2001). The screenshot shows a feedback message (window on top) that refers to some portion of the diagram (highlighted red).

5.1.1 Configuration mechanisms

Figure 23 shows how the configuration settings of LASAD are organized. There are four configuration folders, each containing XML configuration files for one specific aspect: ontologies, templates, users, and maps.

- **`Users`** are defined in terms of user name, password, and role. The role determines the privileges that users have in the system, for instance, access to a debug panel for programmers (*Developer* role) or to a graphical configuration authoring frontend (*Teacher* and *Developer* roles).
- **`Maps`** are defined in terms of a descriptive name (shown in the user interface when users choose which map to join), optional user restrictions (can be used to grant only specific users access to a map), and a template ID, which refers to a template description (see below).
- **`Templates`** define working sets in LASAD in terms of basic configuration settings (e.g., how many users can join a session at a time), available tools (e.g., chat tool or chat tool enhanced with sentence openers), information displays (e.g., a panel displaying which users have joined a session), and an ontology ID, which refers to an ontology description (see below).
- **`Ontologies`** define the elements of the graphical language available in the diagramming space, in particular, the specific box and link types.
- **`Elements`** of an ontology (i.e., available box and link types) are defined in terms of a heading displayed in the user interface, some general properties (e.g., color, size, or whether the box is resizable), and a list of sub-elements the element is composed of (e.g., a text field, radio buttons, a dropdown menu, or an awareness information element). Each sub-element type has its own specific configuration options not detailed here.

Table 28 shows the XML snippet used to define the properties of the box type *Test* within the LARGO ontology (simplified for presentation purposes). Instances of this *Test* box are displayed in Figure 21, p. 157 (boxes number 1 and 2). The XML snippet defines that the box uses a heading *Test* (`heading="Test"`), and has four child elements: three text fields (`elementtype="text"`) with the labels *IF*, *AND*, and *THEN*, and an awareness sub-element (`elementtype="awareness"`). While there is exactly one *if*, *then*, and *awareness* sub-element, the box includes between 0 and 5 *and* sub-elements (`minquantity="0" maxquantity="5"`).

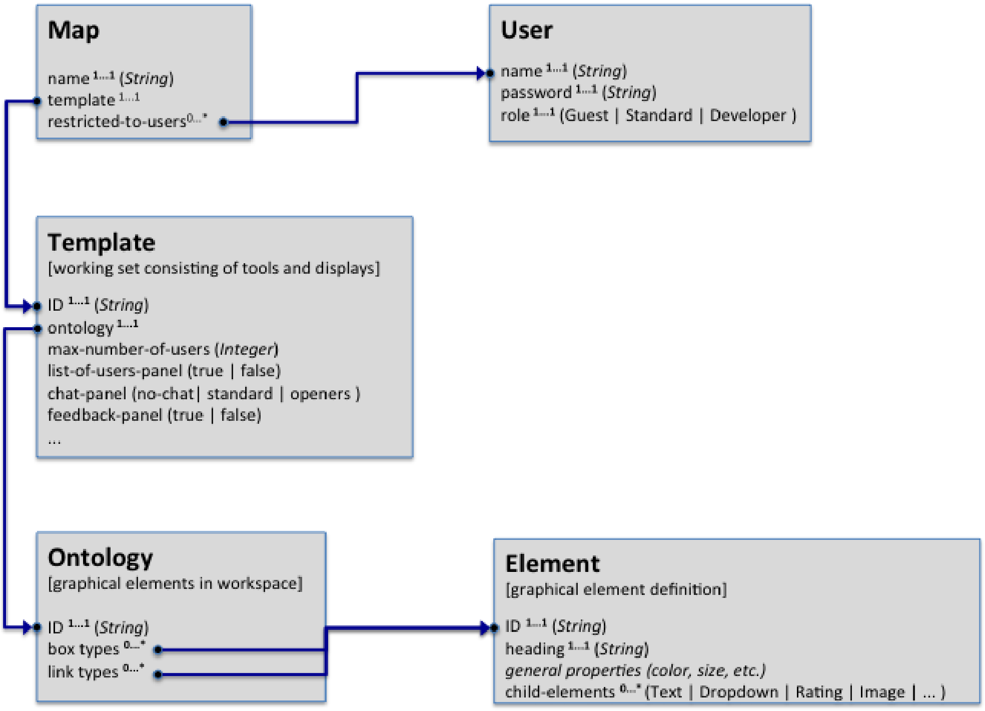

Map
name 1…1 (String)
template 1…1
restricted-to-users 0…*
User
name 1…1 (String)
password 1…1 (String)
role 1…1 (Guest | Standard | Developer)
Template
[working set consisting of tools and displays]
ID 1…1 (String)
ontology 1…1
max-number-of-users (Integer)
list-of-users-panel (true | false)
chat-panel (no-chat| standard | openers)
feedback-panel (true | false)
...
Ontology
[graphical elements in workspace]
ID 1…1 (String)
box types 0…*
link types 0…*
Element
[graphical element definition]
ID 1…1 (String)
heading 1…1 (String)
general properties (color, size, etc.)
child-elements 0…* (Text | Dropdown | Rating | Image | ...)

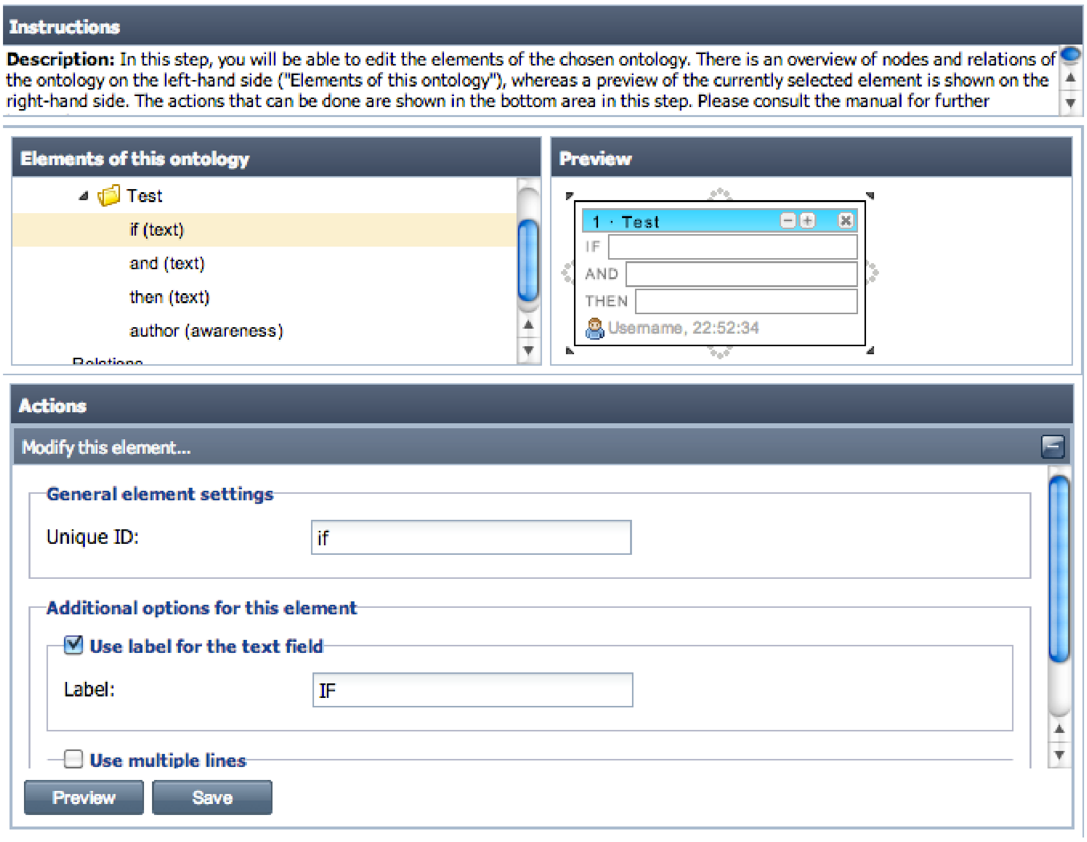
Figure 23: LASAD configuration entities and their relations.

Table 28

Excerpt from the LARGO ontology file showing the XML specification of box type *Test* (simplified)

```
<element elementid="test" elementtype="box">
   <elementoptions heading="Test" />
   <uisettings width="190" height="160" resizable="true"
      border="standard" background-color="#55C3FF"
      font-color="#000000" />
   <childelements>
       <element elementid="if" elementtype="text"
           quantity="1" minquantity="1" maxquantity="1">
          <elementoptions label="IF" text="" />
       </element>
       <element elementid="and" elementtype="text"
           quantity="0" minquantity="0" maxquantity="5">
          <elementoptions label="AND" text="" />
       </element>
       <element elementid="then" elementtype="text"
           quantity="1" minquantity="1" maxquantity="1">
          <elementoptions label="THEN" text="" />
       </element>
       <element elementid="awareness" elementtype="awareness"
          quantity="1" minquantity="1" maxquantity="1">
       </element>
    </childelements>
</element>
```

To support particularly users without a technical background, who might not be familiar with editing XML configuration files, a graphical administration and authoring system has been implemented and integrated with LASAD, allowing users to easily define and administer configuration settings. The LASAD Authoring Tool provides screens to configure ontologies, templates, maps, and user accounts. Figure 24 shows the graphical user interface to define ontologies. The shown configuration is by and large equivalent to the XML snippet in Table 28, that is, it defines a *Test* box composed of three text fields labeled *IF*, *AND*, and *THEN*, and an awareness element.


Instructions

Description: In this step, you will be able to edit the elements of the chosen ontology. There is an overview of nodes and relations of the ontology on the left-hand side ("Elements of this ontology"), whereas a preview of the currently selected element is shown on the right-hand side. The actions that can be done are shown in the bottom area in this step. Please consult the manual for further

Elements of this ontology

- Test
  - if (text)
  - and (text)
  - then (text)
  - author (awareness)
- Relations

Preview

1 · Test

IF

AND

THEN

Username, 22:52:34

Actions

Modify this element...

General element settings

Unique ID: if

Additional options for this element

☑ Use label for the text field

Label: IF

☐ Use multiple lines

Figure 24: Detail of the LASAD Authoring Tool user interface (Loll 2012). The shown screen allows users to define the elements of a LASAD ontology. Adapted from Loll (2012).

5.1.2 Software architecture

Figure 25 shows the overall LASAD system architecture with its principal components, in particular:

- The **`LASAD-Server`** uses a database to maintain the history and state of all LASAD sessions and distributes messages between connected LASAD clients in order to synchronize their states. For instance, when a user creates a new box in the diagram, the graphical user interfaces of all other connected users must be updated as well. Two sorts of LASAD clients exist: `End-User-Clients` and the `CASE-Engine`.

- The **`End-User-Clients`** provide the graphical user interface used by students and teachers. They are JavaScript-based applications, built with the Google Web Toolkit (GWT)[11] and executed in a standard web browser. The `End-User-Clients` also include the LASAD authoring tools, which are only accessible to privileged users. The authoring tools include the "standard" LASAD Authoring Tool discussed in section 5.1 and the LASAD Feedback Authoring Tool, which will be discussed later. Since the `LASAD-Server` cannot directly speak to the JavaScript-based client side, an intermediary `GWT-Servlet` is interposed to mediate the communication between `LASAD-Server` and `End-User-Clients`.
- The **`CASE-Engine`** provides automated analysis and adaptive support functionality to LASAD. From the perspective of the `LASAD-Server`, the `CASE-Engine` is just another client that uses the same infrastructure and interfaces as `End-User-Clients`. Yet, since the `CASE-Engine` is implemented in Java rather than JavaScript, it can directly talk to the `LASAD-Server` through a Java RMI connection without the indirection of the `GWT-Servlet`.

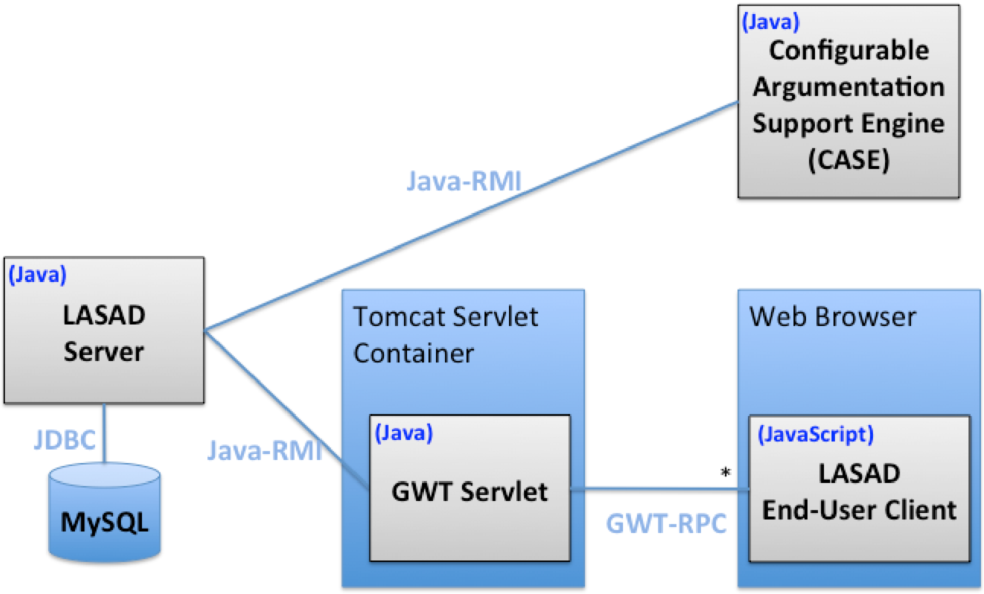


Figure 25: Architecture of the LASAD system. From Scheuer and McLaren (2013).

[11] The Google Web Toolkit (https://developers.google.com/web-toolkit) is a Java-based software development environment, which provides a Java-to-JavaScript compiler to translate Java code into JavaScript code to be executed in web browsers.

To illustrate how LASAD operates, let's have a look at a processing iteration, starting with a user creating a box in the LASAD user interface. The `End-User-Client` then sends a *create box* message through the `GWT-Servlet` to the `LASAD-Server`. The `LASAD-Server` updates its database and forwards the message to all connected clients. `End-User-Clients` display the new box on the screen. The `CASE-Engine` updates its internal data representation, searches for meaningful patterns, and possibly generates feedback messages. These feedback messages are sent through `LASAD-Server` and `GWT-Servlet` to one or multiple `End-User-Clients`, depending on the specific CASE configuration, to be displayed on the screen. While the `CASE-Engine` provides the textual content of feedback messages and control flags, e.g., to specify whether diagram elements should be visually highlighted or not, the actual realization of the feedback presentation is done by the `End-User-Clients`.

5.2 Research Questions

As discussed above, past argumentation systems have been designed with specific domains and learning scenarios in mind, resulting in systems that cannot be easily ported to different application settings. Particularly critical is the development of adaptive support functionality, which involves considerable investments in time and effort. To adaptively support student learning, systems must accomplish at least two tasks: (1) identify relevant patterns in student actions, interactions, and artifacts (the analysis task), and (2) decide, based on the specific pattern and other relevant parameters, if, when, and how to respond (the support task). While the specific patterns of interest and support strategies certainly differ between domains and application settings, there are also many crosscutting aspects—aspects differently targeted applications have in common. For instance, LASAD-based applications typically use node-and-link graphs to represent arguments, no matter which specific domain is tackled. Furthermore, there are typical approaches to present feedback (e.g., highlighting of graphical patterns, canned textual messages) and typical feedback provision strategies (e.g., feedback-on-request, immediate feedback), which are used across systems. So, wouldn't it be possible to develop a general mechanism, or authoring tool, to flexibly define graph-based patterns for arbitrary domains? And wouldn't it be possible to develop a configuration mechanism that allows choosing between the most common and effective approaches to providing feedback? Of course, it is not possible to anticipate all specific requirements future argumentation-learning systems might have. But it should be possible to design an adaptive support system in a way that allows hooking up new modules to extend the existing functionality as needed. In summary, a well-designed software framework may be used to implement adaptive support across domains and learning settings. This chapter will address two research questions:

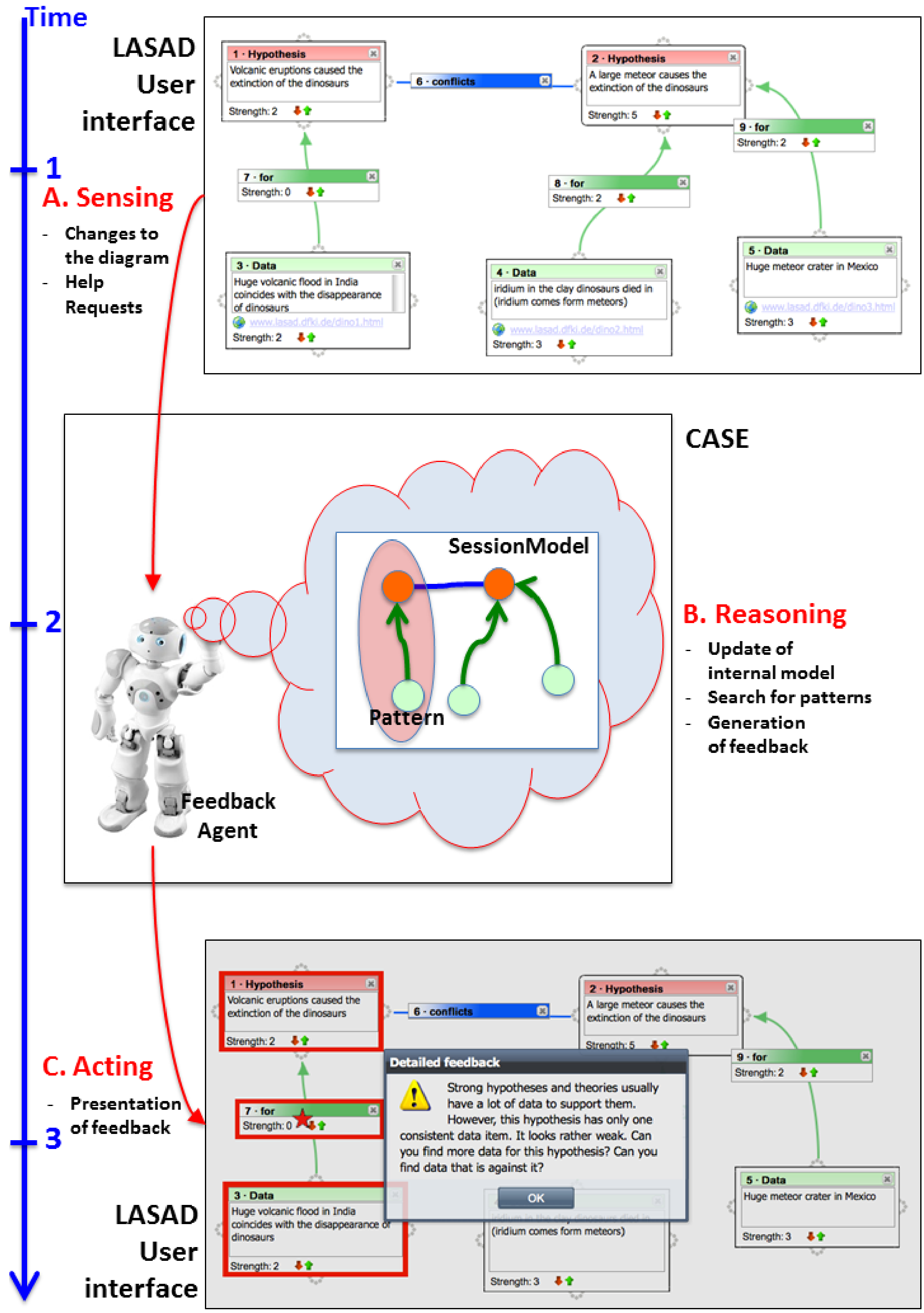
> **(RQ2) Adaptive Support Architecture:** "How can a software architecture be developed to optimally provide adaptive support across different argumentation domains and learning scenarios?"
>
> **(RQ3) Pattern Definition Mechanism:** "How can a pattern search component be developed to optimally support the definition of patterns across different argumentation domains and learning scenarios?"

To the best of my knowledge, this is the first approach that has been developed to address the authoring of adaptive support for argument diagramming activities in a domain-independent manner. Previous approaches always focused on particular argumentation domains. For instance, the Rashi authoring tool (Murray et al. 2004) supports authors in creating expert solutions for science problems, restricted to a model based on hypotheses, data, and their evidential relations. The pattern detection mechanism in the LARGO system was formalized in terms of a graph grammar (Pinkwart et al. 2008)—a formalism also applicable to domains other than legal argumentation, which was the focus of LARGO. Yet, the graph grammar approach was never operationalized into a generalized software framework that flexibly supports the definition and execution of declarative analysis rules.

5.3 Concept

Figure 26 illustrates the basic functioning and important concepts of the CASE engine. The figure depicts a LASAD session (each LASAD session essentially corresponds with one LASAD diagram) with one connected feedback agent. Generally, the CASE engine can maintain an arbitrary number of feedback agents for each LASAD session. Each time students create, modify, or delete elements of the diagram, such as nodes and links, a notification message is sent to the CASE engine to update its internal session model. Feedback agents then search the updated session model for relevant patterns. For instance, the agent in Figure 26 (middle) identified an instance of a pattern consisting of a *Hypothesis* element that is supported by exactly one *Data* element. Typically, feedback agents search for many different patterns and detect many instances at once. For instance, the LARGO agent can identify more than 40 different patterns and sometimes detects more than 100 pattern instances in one diagram (Pinkwart et al. 2006). Each pattern instance essentially represents an opportunity to provide hints and feedback, and is associated with a corresponding canned feedback message. In addition to a textual message, the feedback in Figure 26 (bottom) highlights relevant structures in the diagram. Whether and which feedback message to deliver depends on the agent's specific feedback strategy. The feedback strategy defines, among other things, whether feedback is provided proactively or on request (i.e., the student requests feedback from the LASAD feedback menu), and which feedback to choose if multiple options are available (remember, many pattern instances might be detected at once).

The CASE engine comprises an extensive configuration subsystem, which allows defining and administering feedback agents. Feedback agents are defined on three levels: patterns, feedback messages, and feedback strategies. They can be assigned to specific sessions and ontologies. When assigned to a session, a fresh agent instance will be instantiated for this particular session. When assigned to an ontology, a fresh agent instance will be instantiated for each session that uses this specific ontology.


Time
LASAD User interface
1
A. Sensing
- Changes to the diagram
- Help Requests
1 · Hypothesis
Volcanic eruptions caused the extinction of the dinosaurs
Strength: 2
6 · conflicts
2 · Hypothesis
A large meteor causes the extinction of the dinosaurs
Strength: 5
9 · for
Strength: 2
7 · for
Strength: 0
8 · for
Strength: 2
3 · Data
Huge volcanic flood in India coincides with the disappearance of dinosaurs
www.lasad.dfki.de/dino1.html
Strength: 2
4 · Data
iridium in the clay dinosaurs died in (iridium comes form meteors)
www.lasad.dfki.de/dino2.html
Strength: 3
5 · Data
Huge meteor crater in Mexico
www.lasad.dfki.de/dino3.html
Strength: 3
CASE
SessionModel
Pattern
2
Feedback Agent
B. Reasoning
- Update of internal model
- Search for patterns
- Generation of feedback
C. Acting
- Presentation of feedback
3
LASAD User interface
Detailed feedback
OK

Strong hypotheses and theories usually have a lot of data to support them. However, this hypothesis has only one consistent data item. It looks rather weak. Can you find more data for this hypothesis? Can you find data that is against it?

Figure 26: Conceptual view on the functioning of CASE feedback agents.

5.4 Software Architecture

Figure 27 shows the principal components of the CASE engine:

- The **`DataService`** encapsulates all communication with the LASAD server, translating back and forth between the message format used by the LASAD server and the one used internally in the CASE engine.
- The **`SessionManager`** keeps a record of all sessions that exist on the LASAD-Server and distributes incoming messages to the relevant `Session` abstraction.
- **`Sessions`** comprise all information that is associated with a specific LASAD diagram, including a `SessionConfiguration` (invariable aspects such as the used ontology), a `SessionModel` (fluent aspects such as the current state of the diagram), and `SessionAgents` (processing units employed to analyze and support session activities).
- The **`SessionModel`** serves as a central data repository for `SessionAgents` to access and exchange session-related information. It maintains and continuously updates a representation of the current session state based on the `JessRuleEngine`. It provides an interface to register patterns, which will be searched for during operation, and sends out notification messages when patterns have been detected. The pattern search is executed by two standard components: The `JessRuleEngine` identifies structures in the diagram (e.g., a circular argument). The `AggregationService` evaluates conditions on the number of elements (e.g., more than five nodes of a given type in the diagram). Other approaches to identify patterns can be added through new `SessionAgent` implementations as needed.
- **`SessionAgents`** are processing units that perform specific tasks related to the analysis of sessions and the generation of feedback. At service startup, `SessionAgents` register their patterns of interest at the `SessionModel`. During operation, `SessionAgents` receive notification about detected and invalidated pattern instances from the `SessionModel`. The `SessionAgent` interface provides an extension point in the CASE engine to add new analysis and feedback capabilities. Already existing agents include a configurable `FeedbackAgent` (see above) and the `DeepLoopAgent` (integrates AI-based classifiers to analyze e-discussions).

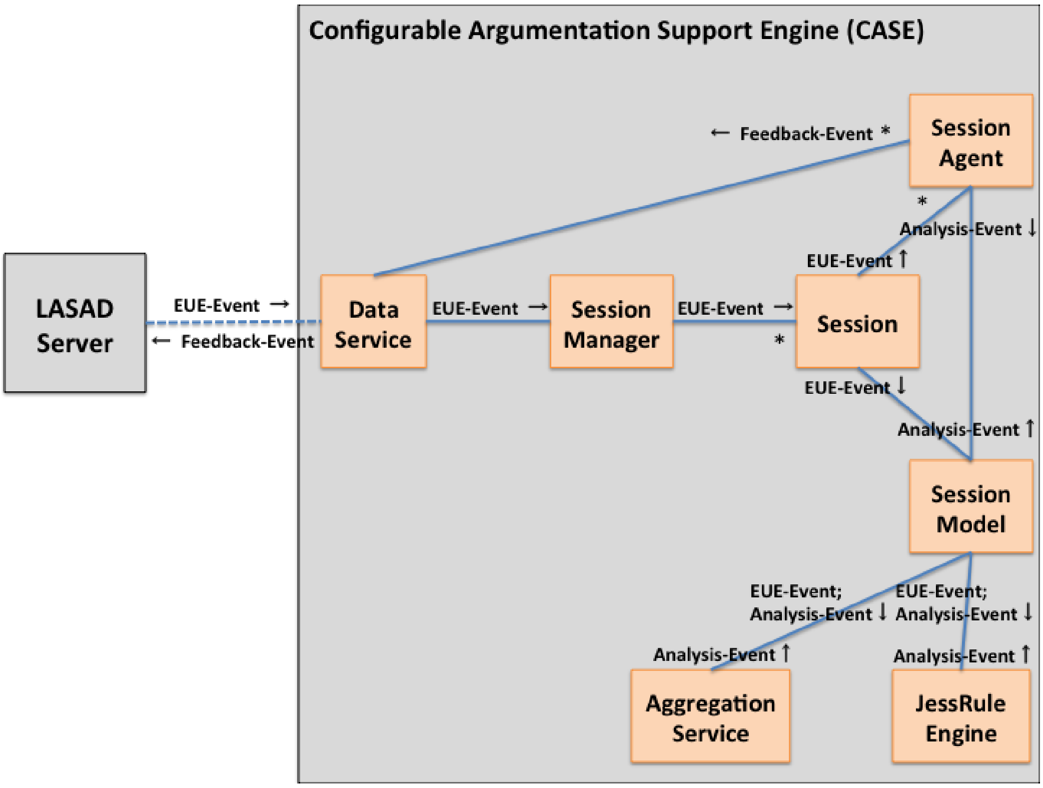


Figure 27: CASE components and interactions. From Scheuer and McLaren (2013).

5.5 Knowledge Representation and Inference

A centerpiece in the CASE architecture is the `SessionModel`, which employs the `JessRuleEngine` to model the current state of LASAD diagrams and to identify salient structures through declarative production rules. This section first gives a short overview of production rule systems (section 5.5.1) before describing the particular approach used in the CASE engine (section 5.5.2).

5.5.1 Background: Production rule systems

Preliminary note: The terms *pattern*, *network*, and *node* used in this subsection are technical terms to explain the general functioning of production rule systems. They should not be confused with patterns, nodes, and networks with respect to argument diagrams.

Production rule systems essentially consist of a fact base, a rule base, and a rule interpreter. The *fact base* (or *working memory*) represents the current world state in terms of discrete knowledge *chunks* (or *facts*). The *rule base* represents inferential knowledge in terms of production rules. A production rule consists of a *left-hand* *side* (LHS), which specifies a pattern (i.e., a certain constellation in the fact base), and a *right-hand side* (RHS), which specifies a sequence of actions (e.g., adding, modifying, or deleting facts). The *rule interpreter* matches production rules against the current world state in the fact base. When the pattern on the LHS is identified in the fact base, the rule is fired, that is, the actions on the RHS will be executed. Typically, an analysis run takes multiple cycles. Production rules modify the fact base, and the changed fact base triggers the execution of production rules again. This data-driven mode of reasoning—repeatedly applying if-then rules (or modus ponens) until no condition can be matched anymore—is referred to a forward chaining. Forward chaining is particularly useful to keep a world model (i.e., the fact base) up to date in a continuously evolving world (i.e., facts are continuously added, removed, or modified). It can be contrasted with backward chaining, a goal-driven mode of reasoning. For instance, the logic programming language Prolog (Merritt 1989) makes use of backward chaining to answer queries by checking whether the conditions of a given conclusion (the query) are satisfied in the current world model.

The Jess rule engine (*Java Expert System Shell*; Friedman-Hill 2003) is a production rule engine that easily integrates with Java applications since it is itself implemented in Java. Jess utilizes the Rete algorithm (Forgy 1982) to efficiently implement forward chaining. The Rete algorithm compiles a hierarchical network from a given rule base. The nodes within the network represent the (logical) tests that are used as part of the pattern definition on the LHS of rules. Child nodes in the network represent tests that are contingent on the results of their parent nodes. For instance, two parent nodes might match an individual fact each, while their common child node matches the combination of both individual facts (i.e., fact1 AND fact2). Nodes at the bottom of the network represent all tests necessary to identify a complete pattern (i.e., the LHS) of a rule. When a new fact is added to the fact base, a corresponding token traverses the network, starting at the top. At each node, the fact is analyzed in terms of the tests attached to that node. Only if the tests are satisfied, the token will be passed through to the child nodes. If a token reaches a node at the bottom of the network, a complete pattern has been matched and the rule is activated (i.e., the condition for executing the actions on the rule's RHS are fulfilled). The trick in the Rete network is that nodes keep a record of facts—and groups of facts—that have successfully passed the test. That is, the network remembers previous test results and only re-computes tests that are necessary due to a change of the world model (i.e., new facts added, existing facts removed). Consequently, the Rete network is particularly efficient if a potentially large world model evolves slowly over time since only a small number of tests must be re-evaluated. Moreover, the Rete network is optimized in that multiple rules share the same nodes in the network if they depend on the same set of tests, that is, the same test is only executed once to evaluate multiple rules. The Jess rule engine has been shown to be a highly efficient implementation of the Rete algorithm. Used on an outdated machine (800 MHz Pentium III, Sun HotSpot JVM) Jess could fire up to 80,000 rules, match up to 600,000 patterns, and add up to 100,000 facts to the Jess knowledge base within one second.

Production rule systems provide a well-suited technical framework to implement expert systems. The basic idea is to emulate human (expert) knowledge and reasoning in a software system. For instance, the probably most widespread class of intelligent tutoring systems, *Cognitive Tutors* (Anderson et al. 1995), utilizes a production rule system to model how experts (and students) solve problems in LISP programming, geometry, and algebra. Similarly, the CASE engine uses the Jess rule engine to emulate human coaches who observe students creating argument diagrams, and identifies pedagogically relevant constellations in these diagrams, e.g., modeling errors to hint at. Typically, each rule identifies exactly one such constellation. In rare cases, multiple rules are required to specify particularly complex constellations. An example Jess rule will be discussed in section 5.6.

5.5.2 Pattern search mechanism of the CASE engine

Figure 28 depicts the specific knowledge representation scheme used in the fact base (blue shaded areas), conversion procedures to translate between Java and Jess-based object representations (orange arrows), and inference procedures to derive new knowledge facts from existing ones (blue arrows).

An incoming user action will be processed within the `JessRuleEngine` as follows. The `SessionModel` translates the user action into an `action` fact and adds it to the fact base. Jess `action` facts represent crucial information about user actions, including the action type (e.g., *create*, *modify*, or *delete*), the actor, a timestamp, and a description of the manipulated object in terms of its ID and semantic type (e.g., *hypothesis*, *data*, *support*, or *oppose*). Depending on whether the manipulated object is a box, a link, or a sub-element (e.g., text field or a dropdown menu), three action subtypes with additional information are defined: `node-actions`, `link-actions`, and `subelement-actions`.

Jess `action` facts are analyzed through a set of Jess rules to reconstruct the current diagram state, represented in terms of `object` facts (i.e., `node`, `link`, `sub-`

element facts). For instance, if the action fact indicates that a node has been deleted, the corresponding node fact will be removed from the fact base. Besides state information, object facts hold relevant process information, such as the object creator, modifiers, and corresponding timestamp information.

Jess object facts are analyzed through application-specific Jess rules to identify patterns of interest in the current diagram state. Since the chosen knowledge representation format of object facts combines structural and process information, structural patterns (e.g., a node *n1* of type *t1* is connected to a node *n2* of type *t2* through a link *l* of type *t3*) can be further constrained through process-related properties (e.g., node *n1* and *n2* were created by different users; node *n1* was created before node *n2*).

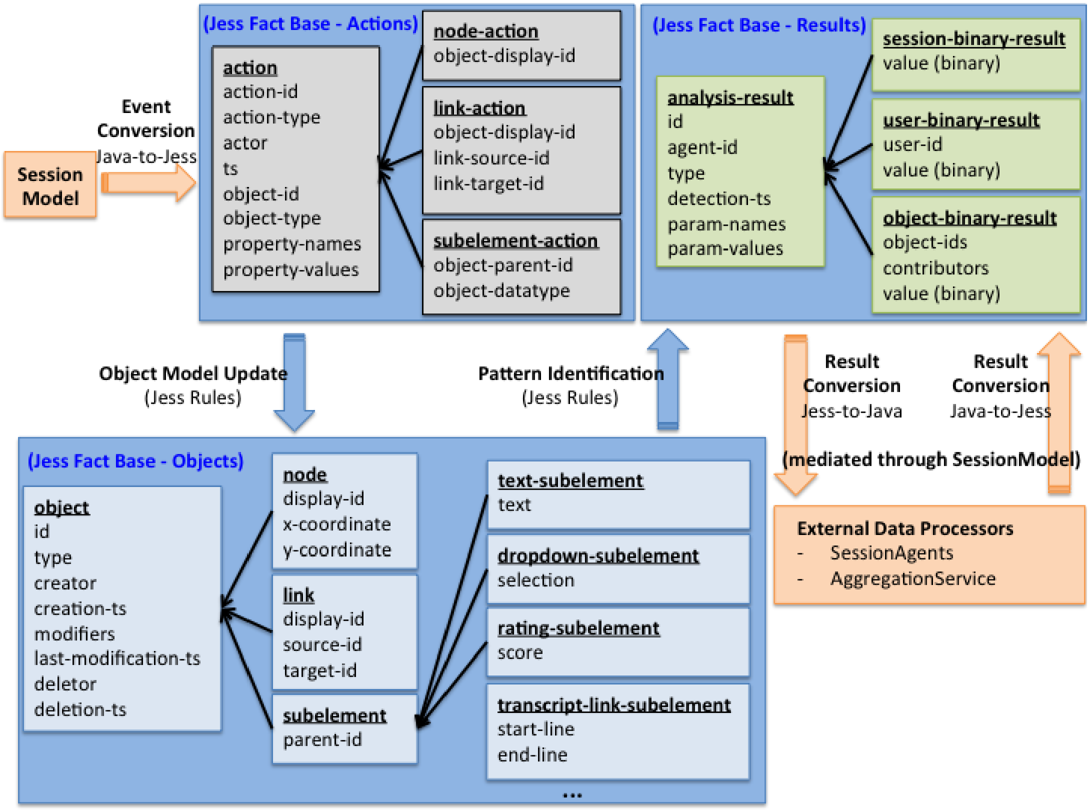


Figure 28: CASE knowledge representation and inference processes. From Scheuer and McLaren (2013).

If a pattern has been identified, a corresponding analysis-result fact is added to the fact base. Patterns might refer to specific sets of objects (object-binary-result), specific users (user-binary-result), or the session as a whole (session-binary-result).

Finally, through a two-way conversion procedure between Jess and Java object representations, external data processors, such as the `AggregationService` and `SessionAgents`, read out and write `AnalysisResult` objects from and to the fact base. For instance, the `DeepLoopAgent` could apply machine-learned classifiers and write corresponding `AnalysisResult` objects to the fact base. The `AggregationService` could read out all `AnalysisResult` objects of a specific type to count how often a certain pattern occurs in a diagram. That is, the analysis approach in the CASE engine is not restricted to rule-based pattern matching operations, but can easily integrate other analysis formalisms (e.g., machine-learned classifiers).

The described knowledge representation scheme can be, and has already been, extended with additional data structures to allow more complex analyses, for instance, by representing tallies, paths, cycles, and predefined expert solutions to specific problem instances (e.g., an expert analysis of a legal transcript that can be compared with the student diagram).

5.6 Configuration Mechanisms

Configuration settings of the CASE engine are maintained in the file system. There are XML configuration files for the following aspects:

- *Connection parameters* specify network address and login credentials to connect to the LASAD server.
- *Agent configurations* specify the behavior of agents.
- *Deployment settings* specify which agent type to deploy to which sessions and ontologies.

At system startup, the CASE engine reads these files, connects to the LASAD server, initializes agent instances, and deploys these agent instances to LASAD sessions. The CASE engine also provides an interface to change configuration settings online, including, adding, modifying, and deleting agent configurations, assigning agents to sessions, and starting and stopping support for sessions. A graphical feedback configuration and administration front-end makes use of this interface (discussed below).

Figure 29 shows how the configuration settings of feedback agents are organized. As mentioned above, feedback agents are just one specific agent type. The CASE engine can be extended with other agent types, which may expose different behaviors and support different configuration parameters, via a generic `Agent` programming interface. The behavior of a feedback agent is defined on three levels.

- **`Patterns`** specify pedagogically relevant situations in argument diagrams. Depending on the specific type of pattern—currently four different types of patterns are supported (see below)—different pattern definition formats are employed. Additional pattern restrictions can be defined, for instance, to limit the scope to recent or old patterns (`min-age` and `max-age` parameters), or patterns the user under consideration has contributed to (`user-restriction` parameter).
- **`Messages`** specify how to respond to specific patterns (`triggering-pattern` parameter). A message consists of a short message (`short-text` parameter), which provides feedback in a concise way, a long message (`long-text` parameter), which provides a more detailed explanation, only displayed when the user clicks on the short message, and a highlighting flag (`highlighting` parameter), which indicates whether objects that are part of the pattern should be visually highlighted in the user interface. The CASE engine implements an approach to classify the current diagram according to usage phases (e.g., an *early*, *main*, and *late* phase). Priority values can be assigned to messages to indicate their relative importance in different usage phases.
- **`Strategies`** specify if, when, and which messages to deliver. Messages might be provided on request only, or periodically, in predefined intervals (`provision-time` parameter). Messages might be delivered to an entire group or to an individual student (`recipient` parameter). Certain messages might be excluded (`filters` parameter). For instance, if the same pattern matches multiple structures in a diagram, only one feedback message may be considered, rather than messages for each pattern instance (`One-Instance-Per-Type` filter). Or messages already delivered in the past might be excluded, based on the history of previous messages (`No-Instance-Twice` filter). The list of messages can be sorted according to a list of predefined criteria (`sort-criteria` parameter), for instance phase-based priorities (`phase-priority` criterion) or pattern age (`prefer-recent-structures` criterion). Finally, a cut-off point for the sorted list of messages must be defined (`number-of-messages` parameter). Through its modular design, the CASE engine can be easily enhanced with further filter and prioritization criteria.

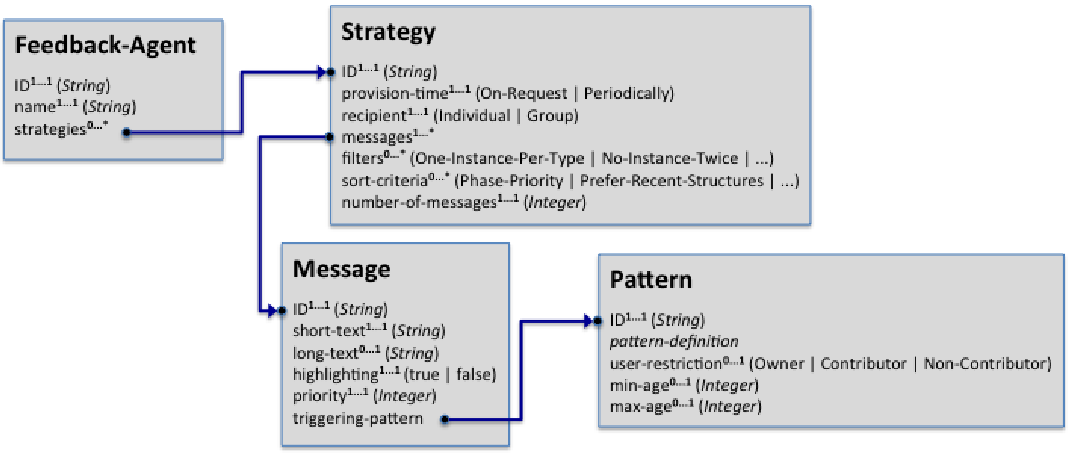

Feedback-Agent
ID$^{1...1}$ (String)
name$^{1...1}$ (String)
strategies$^{0...*}$
Strategy
ID$^{1...1}$ (String)
provision-time$^{1...1}$ (On-Request | Periodically)
recipient$^{1...1}$ (Individual | Group)
messages$^{1...*}$
filters$^{0...*}$ (One-Instance-Per-Type | No-Instance-Twice | ...)
sort-criteria$^{0...*}$ (Phase-Priority | Prefer-Recent-Structures | ...)
number-of-messages$^{1...1}$ (Integer)
Message
ID$^{1...1}$ (String)
short-text$^{1...1}$ (String)
long-text$^{0...1}$ (String)
highlighting$^{1...1}$ (true | false)
priority$^{1...1}$ (Integer)
triggering-pattern
Pattern
ID$^{1...1}$ (String)
pattern-definition
user-restriction$^{0...1}$ (Owner | Contributor | Non-Contributor)
min-age$^{0...1}$ (Integer)
max-age$^{0...1}$ (Integer)

Figure 29: CASE configuration entities and their relations (simplified).

Table 29

XML snipped defining a Jess-Pattern in CASE (simplified)

```
<pattern id="..." type="jess-rule">
   <jess>
      (defrule R123
         (logical
           (node
              (id ?data_id1)
              (type "data"))
           (node
              (id ?hypo_id)
              (type "hypothesis"))
           (link
              (id ?link_id)
              (type "for")
              (source_id ?data_id1)
              (target_id ?hypo_id))
           (not (node
                  (id ?data_id2&~?data_id1)
                  (type "data"))
                (link
                  (source_id ?data_id2)
                  (target_id ?hypo_id))))
          =>
         (assert
           (object-binary-result
              (agent-id "belvedere-coach")
              (type "one-shot-hypo")
              (object-ids ?hypo_id ?data_id ?link_id))))
   </jess>
</pattern>
```

Table 29 shows a pattern definition based on Jess, which corresponds to the pattern in Figure 22, p. 158 (i.e., a hypothesis with exactly one supporting data element). The `pattern` element specifies a pattern ID (`id`) and indicates that the pattern is defined in the Jess rule language (`type="jess-rule"`). In general, the `type` attribute determines how the body of the `pattern` element is interpreted. Accordingly, the other pattern types listed below specify different values for the type attribute. The approach allows CASE to be easily extended to support further pattern types.

The actual pattern definition is enclosed in another XML element (`jess`). The pattern (LHS of the rule) comprises a node of type `data` that supports a second node of type `hypothesis`, indicated by a link of type `for`. A `not` condition specifies that no other `data` node exists that supports the `hypothesis` node. When the pattern on the LHS is matched, the RHS will be executed. In particular, an `object-binary-result` fact that holds important information regarding the detected pattern (agent-id, pattern-id, matched objects) is added to the fact base.

Overall, the CASE engine supports four different pattern types:

- **`Jess-Patterns`** (see example in Table 29) are specified using the Jess rule language. This option offers the full expressive power of the Jess production rule system but also requires basic knowledge about Jess syntax and knowledge representation and understanding of the functioning of rule-based systems more generally. By modifying existing prototypical patterns it should also be possible for non-experts to define patterns of limited complexity without much effort.
- **`Count-Patterns`** are defined in XML and specify conditions on the number of boxes, links, or other patterns in a session (e.g., users with less than five contributions).
- **`External-Patterns`** are analyzed by external components that connect with CASE over a well-defined API. The CASE engine acts as a mere consumer of these patterns, indifferent to how these patterns are defined or computed, so there are also no restrictions in this respect (e.g., machine-learned models can be used).
- **`XML-Patterns`** are based on a XML language developed to reduce the complexity inherent in the original Jess rules. The goal is to achieve a favorable tradeoff between expressiveness and ease of use. `XML-Patterns` are automatically translated into operational Jess code.

5.7 Showcase Applications

The generality and breadth of applicability of the CASE engine, the main objective and driving force in the design of the system, has been demonstrated with four CASE applications (LARGO, Science-Intro, Metafora, and Argunaut). These applications support argumentation-learning activities in different domains (the Law, science, group deliberation, and ethical discussion), focus on different argumentation facets (analysis of arguments, planning of arguments, and argumentative discourse), and use different features of the CASE engine (structural patterns, process-based patterns, and integration of external analysis modules).

In LARGO, students analyze and structurally represent legal argumentation processes using argument diagrams. In Science-Intro, students use diagrams as an outlining tool to prepare the writing of research reports in the domain of psychology. Both applications are primarily designed for single-user activities. Adaptive support is provided on request and based on structural patterns defined by domain experts. In Metafora, students jointly work in an inquiry environment for mathematics and science. They use LASAD diagrams to discuss, in a structured way, findings obtained in microworld simulations, with the aim of arriving at a joint, agreed solution. In contrast to LARGO and Science-Intro, in which the CASE engine is used to detect domain-specific structures in diagrams, the focus is on interaction patterns to support students in "learning to learn together." Argunaut also focuses on interaction patterns but uses a different analytical approach. Rather than relying on expert-defined patterns, machine-learned classifiers are utilized to categorize qualitative aspects of e-discussions about controversial ethical dilemmas.

For each application, a number of patterns in argument diagrams have been identified, which can be used as opportunities to support students (or teachers) with feedback. Figure 30 shows an example pattern for each application. Table 30, p. 178, summarizes the specific configuration settings of the four previously discussed CASE applications.

– The *LARGO pattern* (upper left) consists of a circular structure of nodes, in which each node leads to or is modified to the next node. The semantics of a *leads-to* or *modified-to* transition often involve a temporal progression, which is counteracted by the pattern's circularity. However, if interpreted as logical consequence, a circular structure can make sense. This pattern can be used to prompt students to rethink their diagram model (temporal or logical relation?) to identify possible mistakes.

– The *Science-Intro pattern* (lower right) consists of a *Hypothesis* node with fewer supporting than opposing inbound links. In general, it is good if students also consider evidence that contradicts a hypothesis rather than only searching for confirmatory evidence, a well-documented psychological phenomenon (*confirmation bias*). However, sometimes students also neglect supporting evidence, or might neither pay attention to supporting nor to opposing evidence. This pattern identifies such situations in order to prompt students to search for positive evidence. Confirmation bias, i.e., neglecting contradictory evidence, could be detected analogously.

- The *Metafora pattern* (lower left) consists of a *Help-Request* node, not older than 10 minutes, unattended for more than 3 minutes (i.e., 3 minutes passed by and no other box has been connected to the help request). It indicates that a student requested help regarding a problem encountered in a microworld. Yet, the help request went unnoticed, or is deliberately ignored, since three minutes have passed without a response. Because the help request is still recent—it has been published within the last ten minutes—it might be worthwhile to draw the attention of other students to this request in order to elicit help.
- The *Argunaut pattern* (upper right) consists of off-topic contributions, identified through a machine-learned classifier. A few off-topic contributions may be acceptable but if an e-discussion goes astray, with a considerable number of contributions not addressing the topic at hand, a human or artificial moderator might want to intervene.

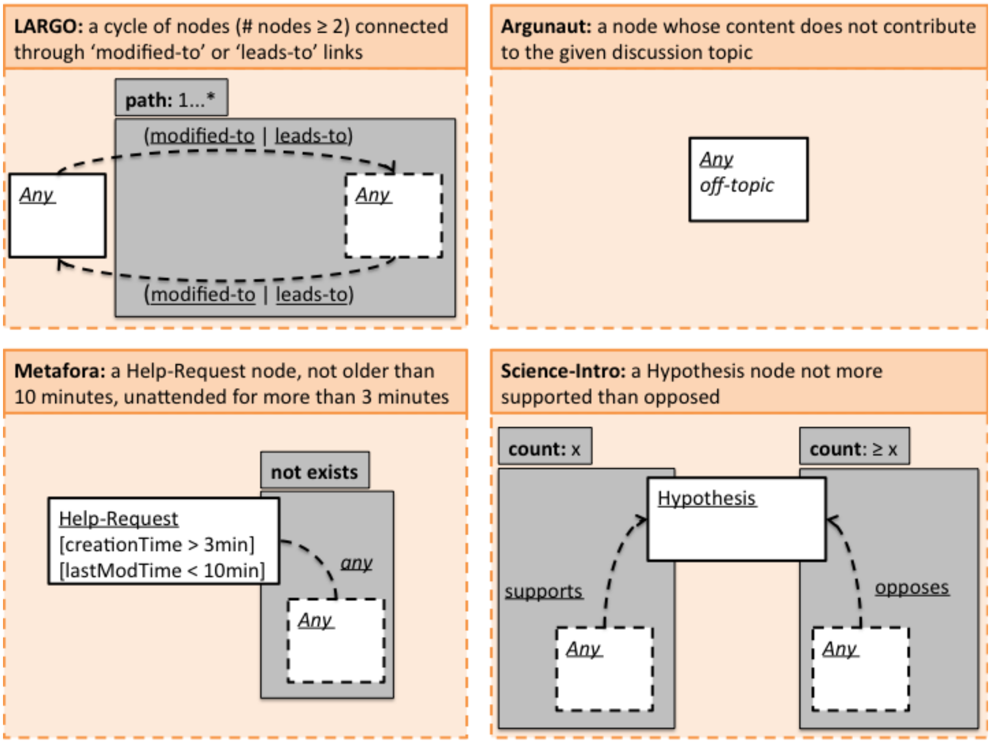

LARGO: a cycle of nodes (# nodes ≥ 2) connected through 'modified-to' or 'leads-to' links
path: 1...*
(modified-to | leads-to)
Any
Any
(modified-to | leads-to)
Argunaut: a node whose content does not contribute to the given discussion topic
Any
off-topic
Metafora: a Help-Request node, not older than 10 minutes, unattended for more than 3 minutes
not exists
Help-Request
[creationTime > 3min]
[lastModTime < 10min]
any
Any
Science-Intro: a Hypothesis node not more supported than opposed
count: x
count: ≥ x
Hypothesis
supports
opposes
Any
Any

Figure 30: Example patterns used in four different CASE applications: LARGO, Science-Intro, Metafora, and Argunaut.

Table 30
Configuration settings of different CASE applications: LARGO, Science-Intro, Metafora, Argunaut

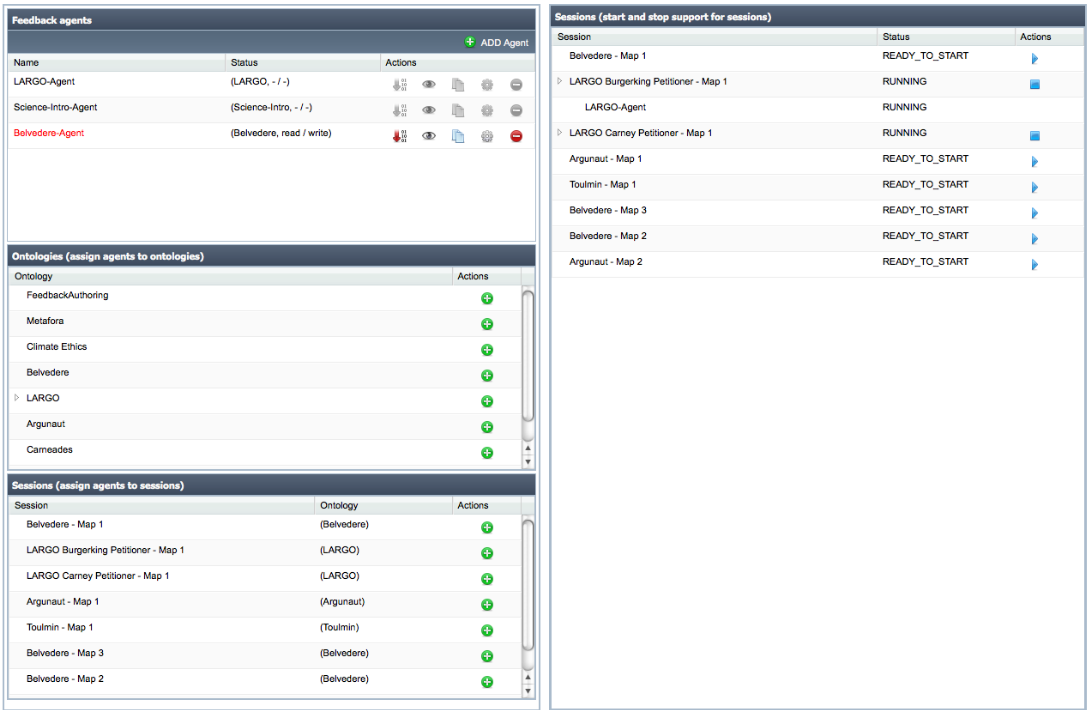
Application	Patterns	Actions	Strategy
LARGO (analysis of legal argument. transcripts)	– *Count-Patterns* focusing on task progress (e.g., no relations in diagram but at least 3 nodes; no "Test" nodes in diagram) – *Jess-Patterns* focusing on domain structures (e.g., a "Hypothetical" node isolated from "Test" and "Fact" nodes; a "Test" node without text in the "Condition" text field) – *Jess-Patterns* focusing on problem-specific aspects represented in an expert model (e.g., important text passages not yet included in diagram)	– *Text message* focusing on problem-solving support – *Highlighting* of diagram elements	– *On-Request* – *Delivered to* the requestor – *Prioritize* based on current problem-solving phase – *Select top-5* hints – *Filter out* all but one message per type
Science-Intro (preparation for writing argumentative texts in science classes)	– *Jess-Patterns* focusing on domain structures (e.g., unconnected node clusters ["argumentation islands"])	– *Text message* focusing on problem-solving support – *Highlighting* of diagram elements	– *On-Request* – *Delivered to* the requestor – *Prioritize* based on current problem-solving phase – *Select top-5* hints – *Filter out* all but one message per type
Metafora (group deliberation about science and math problems)	– *Jess-Patterns* focusing on process characteristics (e.g., unattended help requests)	– *Text message* focusing on collaboration support – *Highlighting* of diagram elements	unattended help requests: – *Automated* – *Delivered to* the entire group – *No prioritization* – *Select* one message – *Filter out* instance already pointed to
Argunaut (argument. about ethical controversies)	– *External-Patterns* (analyzed by machine-learned classifiers) focusing on process characteristics (e.g., off-topic contributions; question-answer pairs)	– *Highlighting and labeling* of diagram elements to support the awareness of moderators regarding salient events	– *On-Request* (classifiers can be invoked separately) – *Delivered to* the requestor – *No prioritization* – *Select* all messages – *No filters*

Note: Adapted from Scheuer and McLaren (2013).

5.8 Graphical Adaptive Support Authoring Frontend

The definition of feedback agents using XML and the Jess rule language involves considerable time and efforts. Users have to know about the general syntax of XML, and perhaps even more critically, they must be able to program Jess rules, a skill that most people have to learn first. On top of that, they must know about the specifics of the CASE configuration mechanism, that is, the data model used in Jess, and the configuration options and their arrangement in the XML files, which essentially constitute a language on their own. The whole process is error-prone and requires extensive testing to avoid syntactical errors and mistyped reference IDs, and to ensure that the feedback agent finally behaves as expected. To ease some of these problems, a graphical authoring frontend was designed, which allows configuring and administering feedback agents using the graphical user interface of LASAD. The Feedback Authoring Tool enables users to manipulate important aspects of CASE during runtime without the need to restart the CASE engine, for instance, defining new feedback agents, assigning feedback agents to sessions, and starting (or stopping) agent support for sessions. It frees users from learning Jess and XML, prevents syntactical errors, and provides guidance through the process of configuring feedback agents. However, the kind of patterns that can be specified is somewhat restricted. For instance, iterative patterns (e.g., node chains and cycles of arbitrary length) cannot be defined with the Feedback Authoring Tool. Technically, this would not be a problem. However, such more advanced constructs would increase the complexity of the tool considerably.

Figure 31 shows the main administration screen of the Feedback Authoring Tool. The panel in the upper left shows all feedback agent types that are available. Agents highlighted red are still under construction and cannot be used yet. Different actions are available, for instance, to create a new agent, to duplicate an existing agent (e.g., to create different but similar agent versions), to change the configuration of an agent, or to delete agents. The panels in the middle and lower left allow assigning agents to sessions and ontologies (i.e., the agent will be assigned to all sessions with this ontology). The panel on the right shows all sessions available on the LASAD-Server. Users can start support for sessions or stop support for sessions (i.e., all agents assigned to the session will be deployed or un-deployed).



Feedback agents

⊕ ADD Agent

Name	Status	Actions
LARGO-Agent	(LARGO, - / -)	
Science-Intro-Agent	(Science-Intro, - / -)	
Belvedere-Agent	(Belvedere, read / write)	

Ontologies (assign agents to ontologies)

Ontology	Actions
FeedbackAuthoring	⊕
Metafora	⊕
Climate Ethics	⊕
Belvedere	⊕
LARGO	⊕
Argunaut	⊕
Carneades	⊕

Sessions (assign agents to sessions)

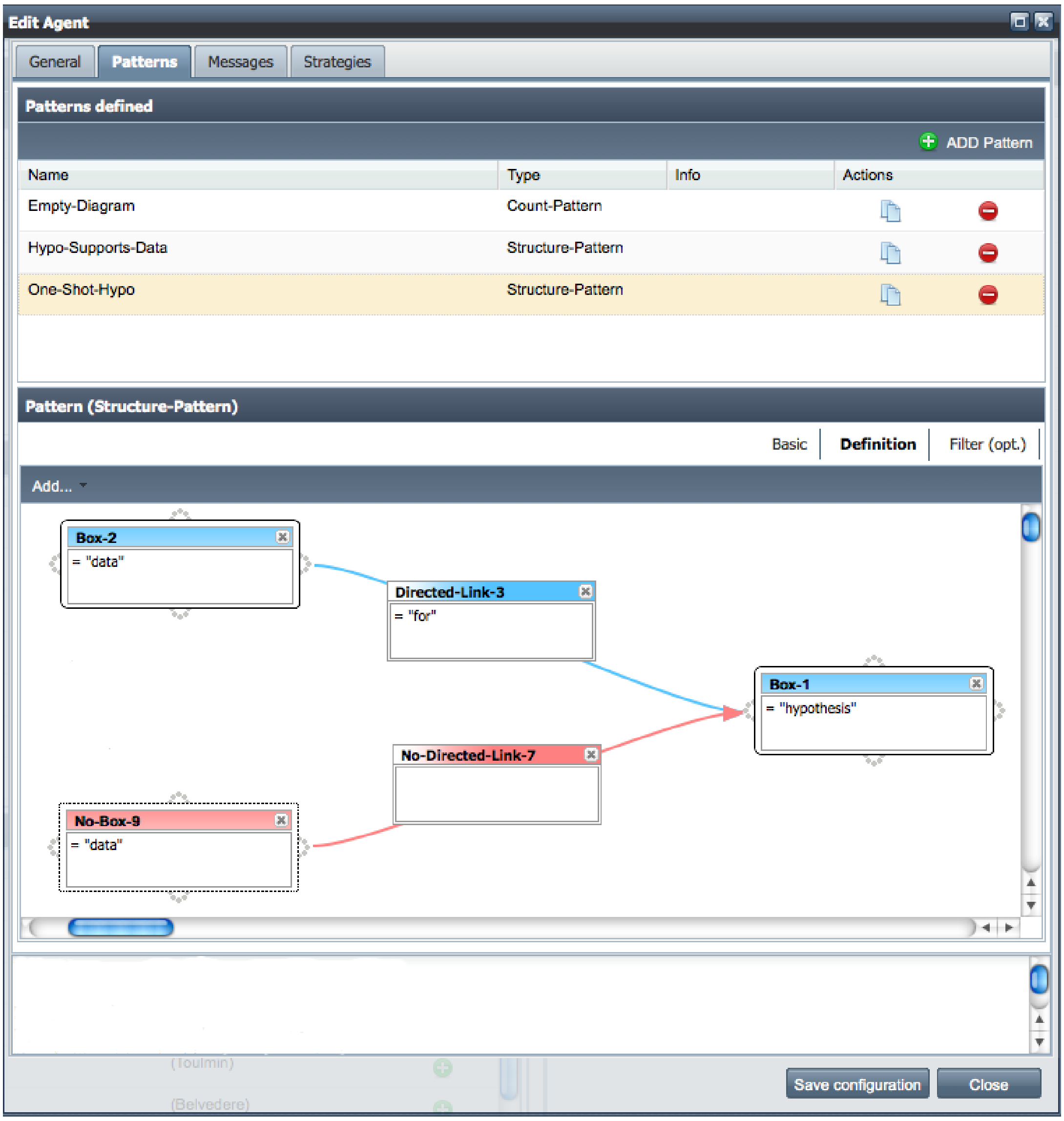
Session	Ontology	Actions
Belvedere - Map 1	(Belvedere)	⊕
LARGO Burgerking Petitioner - Map 1	(LARGO)	⊕
LARGO Carney Petitioner - Map 1	(LARGO)	⊕
Argunaut - Map 1	(Argunaut)	⊕
Toulmin - Map 1	(Toulmin)	⊕
Belvedere - Map 3	(Belvedere)	⊕
Belvedere - Map 2	(Belvedere)	⊕

Sessions (start and stop support for sessions)

Session	Status	Actions
Belvedere - Map 1	READY_TO_START	▶
LARGO Burgerking Petitioner - Map 1	RUNNING	■
LARGO-Agent	RUNNING	
LARGO Carney Petitioner - Map 1	RUNNING	■
Argunaut - Map 1	READY_TO_START	▶
Toulmin - Map 1	READY_TO_START	▶
Belvedere - Map 3	READY_TO_START	▶
Belvedere - Map 2	READY_TO_START	▶
Argunaut - Map 2	READY_TO_START	▶

Figure 31: Feedback agent administration screen of the CASE Feedback Authoring Tool.

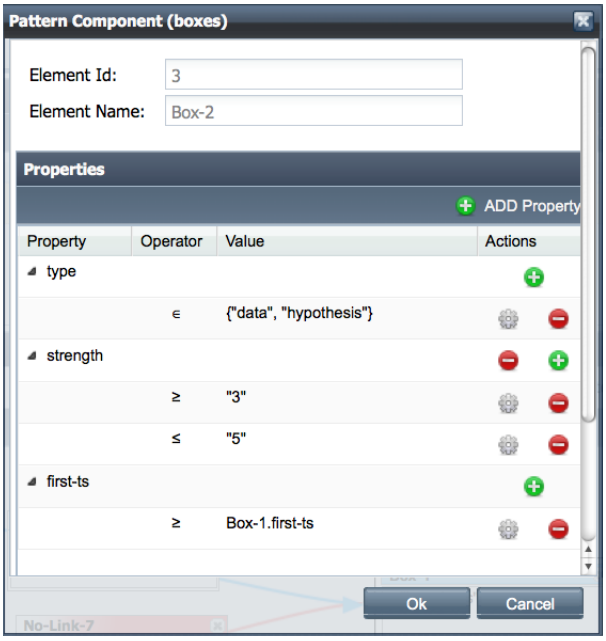
Figure 32 shows the pattern definition screen of the Feedback Authoring Tool (similar screens are available to define messages and strategies). The upper panel shows the list of patterns that have already been defined for this feedback agent. The lower panel shows the definition of one selected pattern. User can use the standard graphical language of LASAD to define the structure of patterns. This specific instance shows the graphical equivalent to the pattern defined in Jess in Table 29, p. 174 (i.e., a hypothesis with exactly one supporting data element). Blue colored elements indicate existing structures. Here, a *data* box (`Box-2`) points to a *hypothesis* box (`Box-1`) via a link of type *for* (`Directed-Link-3`). Red colored elements indicate structures that must not exist. In this example, the pattern indicates that there is no second *data* box (`No-Box-9`) that points to the *hypothesis* box through an arbitrary link (`No-Directed-Link-7`). The patterns are transferred, together with the complete feedback agent configuration, to the CASE backend, and there translated into XML (for persistent storage) and Jess code (to search for the specified patterns in maps).



Name	Type	Info	Actions
Empty-Diagram	Count-Pattern		
Hypo-Supports-Data	Structure-Pattern		
One-Shot-Hypo	Structure-Pattern		

Figure 32: Pattern definition screen of the CASE Feedback Authoring Tool.

Figure 33 shows the constraint definition dialogue of the Feedback Authoring Tool. This dialogue supports the definition of constraints for boxes and links that are part of a pattern, for instance, to specify that a box be of type *data*. Users can bring up this dialogue by clicking on an *edit* icon in the title bar of boxes and links (only displayed when hovering the mouse pointer over a box, therefore not shown in Figure 32). The example in Figure 33 specifies that `Box-2` must be of type *data* or *hypothesis*, the strength rating element within `Box-2` has been set to a value between *3* and *5*, and `Box-2` has been created after `Box-1`, which is another component of the pattern (`first-ts > Box-1.first-ts`).


Pattern Component (boxes)

Element Id: 3

Element Name: Box-2

Properties

Property	Operator	Value	Actions
type			
	∈	{"data", "hypothesis"}	
strength			
	≥	"3"	
	≤	"5"	
first-ts			
	≥	Box-1.first-ts	

Figure 33: Constraint definition dialogue of the CASE Feedback Authoring Tool.

The tool was evaluated in a small-scale study (*n* = 16) with students and academic staff of different disciplinary background (education, psychology, computer science) at Saarland University (Valero Haro 2013). The study focused on the questions how easy it is (1) to use the tool (usability) and (2) to learn how to use the tool (learnability). The hypotheses investigated were (1) that the Authoring Tool is generally easy to use, (2) that the definition of patterns is, due to its inherent technical complexity, the most complicated aspect of the tool, and (3) that technical expertise is an important factor with respect to the ease with which the tool can be used and learned.

We elicited relevant information regarding the participants' background in a pre-questionnaire (familiarity with LASAD and other argumentation diagramming software, knowledge in programming and artificial intelligence, experience in teaching argumentation). Then, participants watched a tutorial video about LASAD and the Authoring Tool (25 minutes). Then, participants worked on 12 differently targeted tasks with the Authoring Tool (more about that below). The performance scores of the participants served as the basis for assessing ease of use and learnability in an objective way. Finally, participants filled in a post-questionnaire, in which they subjectively assessed the usability and learnability of different aspects of the Authoring Tool. In addition, the post-questionnaire gave participants the opportunity to provide further feedback and comments in an open format.

Two groups of tasks were employed:

– *agent administration tasks* (i.e., creating agents, assigning agents to sessions and ontologies, starting and stopping support for sessions), and
– *agent configuration tasks* (i.e., defining patterns of varying complexity, defining feedback messages, defining feedback strategies).

In the data analysis, each task group was evaluated on three different dimensions: "objective task performance" (number of errors), "subjective usability" (5-point Likert scale), and "subjective learnability" (5-point Likert scale).

Figure 34 shows the participants' average usability and learnability ratings. The Feedback Authoring Tool was highly rated, with average scores above 4 (out of 5) for each of the four tested aspects (i.e., usability administration, usability configuration, learnability administration, learnability configuration). This is also reflected in the comments of several participants in the post questionnaire, e.g., that "learning to use this tool is quite simple," "the software is [...] easy to use," "[it] is a useful tool," or simply, "a great tool."

Figure 35 shows the results for the objective task performance. The analysis revealed that participants did not have problems in accomplishing administrative tasks (only one critical error across all participants). Likewise, configuration tasks not involving patterns, such as defining messages and strategies, did not pose a problem for participants (no errors at all). However, participants struggled a lot with the definition of patterns. On average, participants solved only 55% of the given pattern-related tasks without critical errors ($M = 0.55$; $SD = 0.17$). So, the hypothesis that the Authoring Tool is easy to use (hypothesis 1) is at least partially confirmed.

We already anticipated possible problems with some aspects of the tool in hypothesis 2, which predicted that the definition of patterns is the most difficult aspects of the tool. The above described results confirm this hypothesis. Defining patterns requires abstract thinking and generalizing, since each pattern represents an entire class of possible instances rather than an individual instance. Furthermore, it requires proficiency in concepts of logic to specify constraints in patterns. For instance, a simple constraint in Figure 32, p. 181, is that `Box-2` is of type *data* (`Box-2.type = "data"`). More complex constraints may involve variable bindings (e.g., two boxes of an arbitrary but identical type), set concepts (e.g., a box type is element of the set *{data, hypothesis}*), logical connectors (e.g., a box contains no text *and* is of type *data*), and logical negation (e.g., a box does *not* exist; a box is *not* of a certain type). According to verbal reports, particularly the last mentioned aspect—

logical negations—caused confusion. It was difficult for some participants to distinguish between a *data* box that does not exist, and a box that is not of type *data*. One participant explicitly commented in the post questionnaire on possible difficulties in defining logical conditions: "logical operators maybe not clear for everybody." Another participant identified implicitly the definition of constraints as the most critical aspect: "most things are quite intuitive if you have an understanding of what constrains/patterns are."

A reasonable assumption is that participants with a strong background in disciplines that involve a fair amount of formal logic will perform better in pattern-related tasks (hypothesis 3). And indeed, this hypothesis was confirmed at well. There was a strong, significant correlation between programming experience (elicited in a pre-questionnaire) and task performance ($r = .61$; $p = .01$).

Overall, participants performed quite reasonable, given the limited amount of exposure to the tool they had (less than 90 minutes), the stressful situation of a study (being observed, providing think-aloud verbal reports during task execution), and the complexities inherent to the concept of a pattern discussed above. With more training and routine, it is certainly possible to use the Feedback Authoring Tool much more effectively and efficiently. Along these lines, one participant commented in the post questionnaire: "It is a useful tool that needs time and a bit of patience to familiarize with."

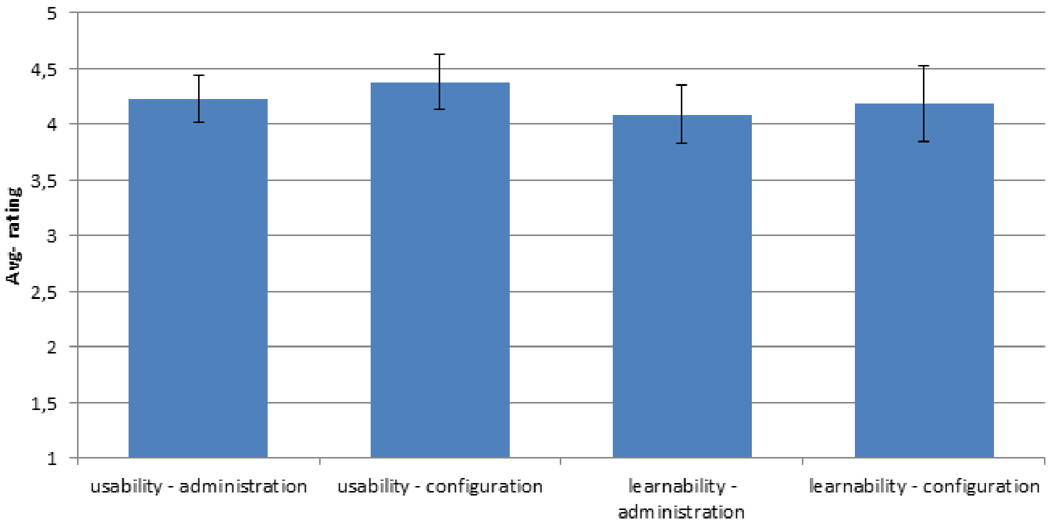


Figure 34: Average rating for the usability and learnability of the configuration and administration functions. Error bars indicate 95% confidence intervals.

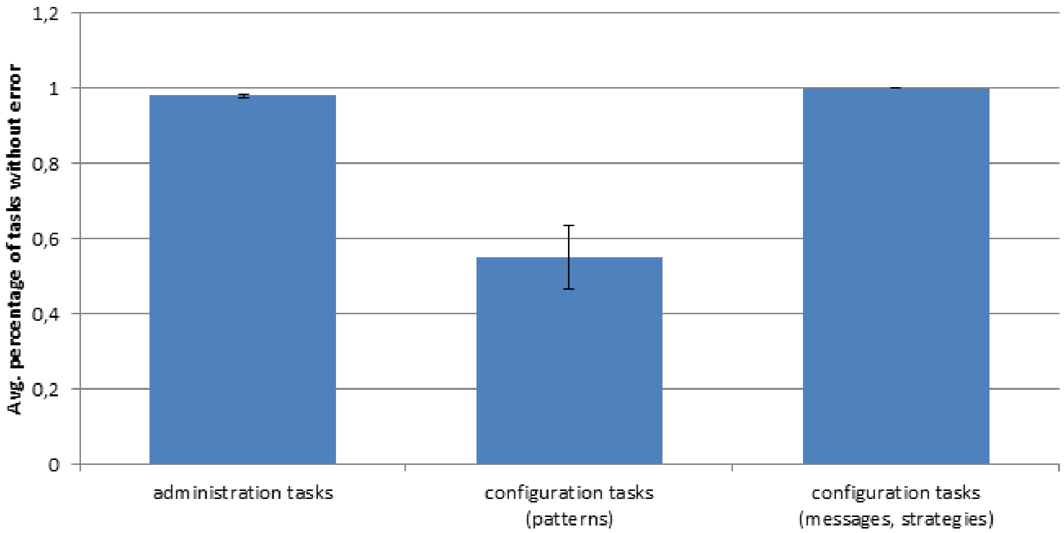


Figure 35: Average percentage of tasks performed without errors, grouped by task category. Error bars indicate 95% confidence intervals.

5.9 Discussion and Conclusion

The CASE engine is a highly configurable software component to analyze and support educational argument diagramming activities. The CASE architecture has been devised with important software design considerations in mind. Maintainability and extensibility have been achieved through a modular design and predefined extensions points, which enable new functionality to be easily added. In order to make the CASE engine highly configurable and thus usable across a wide range of scenarios and domains a comprehensive configuration subsystem was created, parameterizable through XML and a dedicated API, allowing configuration changes at any time. The built-in mechanisms for parameterizing feedback agents enable researchers and practitioners to enhance a wide spectrum of applications with adaptive support functionality. To illustrate this, four applications are presented, demonstrating the diversity of the CASE engine in terms of different argumentation domains, student tasks, and types of support. To facilitate the development of adaptive support functions, and make the configuration framework more accessible for technical novices, the CASE engine has been enhanced with a graphical frontend. An evaluation study yielded promising results.

The CASE engine is the first approach that allows the domain-independent authoring of automated adaptive support functions targeted at argument diagrams. The CASE engine also differs in other important ways from previous approaches. While systems like Belvedere (Suthers et al. 2001), Rashi (Dragon et al. 2006), and

LARGO (Pinkwart et al. 2009) focus on patterns rooted in their respective knowledge domains, the CASE engine can also detect patterns of collaboration (e.g., students who contributed only little to a group diagram; students who did not interrelate their contributions to contributions of fellow students). Moreover, the CASE engine easily integrates with external analysis modules, and offers a broad set of options to define feedback messages and strategies on top of the pattern definition framework. Finally, the CASE engine is the first system that provides a graphical language to define argumentation pattern in a format that can be automatically translated into operational analysis code. In fact, I'm not aware of any graphical language to define "operational" patterns in graphs—including areas other than argumentation. So the presented approach may transfer well into other areas in which intuitive interfaces to search graphical patterns in graph-based data representations would provide a value added.

Harrer et al. (2007) present an approach that is superficially similar. They developed a user interface that allows researchers to specify patterns in argument diagramming activities. Yet, in contrast to the CASE approach, their approach focuses on the sequence of activities (e.g., create box, enter text in box, move box, edit text in box, etc.), and not on the diagrams that emerge from these activities. In fact, they made the observation that researchers were often particularly interested in graphical patterns in diagrams, which cannot be easily defined in their tool, but which are well supported in the CASE Feedback Authoring Tool.

An empirical study underlined the potential of the CASE Feedback Authoring Tool—participants rated the system's usability and learnability highly and performed reasonably well in practice, given the limited instruction time and virtually no training. The study also showed important limitations and potential areas of future research. Particularly those participants without a computer science background struggled with the specification of logical constraints for patterns. Future research may address this potentially critical barrier for users without formal training in logics by investigating the suitability of approaches like *specification-by-example* (Harrer et al. 2007; cf. Zloof 1977). In this approach, users provides a single or a small number of concrete examples of a pattern and the system tries to automatically infer corresponding constraints. Chapter 6 discusses an in some respect related approach, supervised machine learning, which generalizes from a large set of examples (rather than a small one). Of course, automatically inferring constraints from only a few examples is a highly underspecified process and thus must use heuristics that may lead to undesired results. For instance, if the user provides one example box of type *A* and a second example box of type *B*, does this mean that the pattern requires the box to be of type *A* or *B* (and nothing else), or does it mean that the box can be of any type (that is, the example boxes were only coincidantially of type *A* and *B*)? Therefore, the process of inferring patterns from examples could be informed through an interactive dialogue with the user, that is, the system may ask the user questions to tailor the constraints to the specific user requirements and to resolve ambiguities.

While the CASE engine is currently focused on support for argument diagramming activities, the provided framework can easily be extended to also provide support for chat discussions. Chat events, as well as the infrastructure to provide feedback to chat discussions, are already available in the CASE engine. What is missing is an appropriate approach to detect patterns in chat messages and chat sequences. The current approach to manually define pattern-matching production rules appears to be impracticable when it comes to identifying complex structures in natural language—the next chapter discusses an approach still feasible in such situations. In particular, a machine learning approach to analyze natural language texts in argument diagrams is presented. A similar approach may be used to build analytical models for patterns in chat conversations as well.

In summary, the proposed solution provides a proof-of-concept to answer the research questions presented at the outset of this chapter, which were seeking for a software architecture (RQ-2) and pattern search approach (RQ-3) to enable the flexible definition of adaptive support mechanisms *across* argumentation domains and learning scenarios. That the approach has principially accomplished the defined goals is shown, by demonstration, based on four showcases. Of course, to gauge its true potentials, limitations, and possibilities for improvement, practical application on a broader scale would be required. Moreover, the advances and extended possibilities in relation to existing approaches have been discussed, clarifying the contributions made beyond the current state of the art. Many things are “in principle” possible with a sufficiently open technical design, so an important question is whether it can be realistically expected that target users (i.e., designers of adaptive support in argumentation learning systems) are actually able to realize potential benefits. Encouraging evidence that CASE does, in fact, fullfil this important requirement has been collected in the presented usability evaluation study.

Chapter 6

Using Machine Learning Techniques to Analyze Student Discussions

As long as patterns can be described in terms of their graphical structure, node and link types, and simple constraints on other properties, their manual definition is a manageable task. However, not all patterns can be expressed in terms of the explicit semantics encoded in the diagrams. For instance, one may want to detect diagram elements with content not relevant to the current topic (off-topic contributions), or diagram elements with an assigned type not corresponding to the actual type (e.g., a box of type *data* but with text formulating a hypothesis). In these situations, one must identify patterns in the natural language text of diagram elements. Human experts could manually define lists with positive and negative keyword indicative of, for instance, off-topic contributions. This is technically possible within the framework described in the previous chapter, but begs the question of how to come up with a reasonable list of keywords. Moreover, simple keywords are sometimes not sufficient, because the pattern of interest may involve composite terms or grammatical structures, which again dramatically increases the complexity of manually defining the pattern. The involvement of natural language is only one example for situations, in which the manual definition of patterns may become an intractable task. In many situations, humans are able to classify objects, persons, or events without any problem, but struggle when it comes to describing or explaining the rules and criteria these decisions are based upon. In the words of Dahlbom and Mathiassen (1993, p. 33, italics mine):

> "We have no idea *how* we do a lot of the things that we know how to do. Among those are the very fast feats of perception, recognition, attention, information retrieval, and motor control. We know how to see and smell, how to recognize a friend's face, how to concentrate on a mark on the wall or search memory for an old experience. [...] These are definitely *tacit competencies*. If there are rules involved, we have no idea what they might be."

While the *top-down* (or *knowledge-driven*) approach meets its limits here, *bottom-up* (or *data-driven*) approaches may constitute a viable alternative. Machine learning techniques can be used to automatically infer patterns from a given set of examples, making a manual pattern definition obsolete. For instance, one could provide positive and negative off-topic examples (i.e., off-topic and on-topic contributions). A machine-learning algorithm could automatically analyze the given examples to learn a computational model that can distinguish between off-topic and on-topic contributions with reasonably high accuracy.

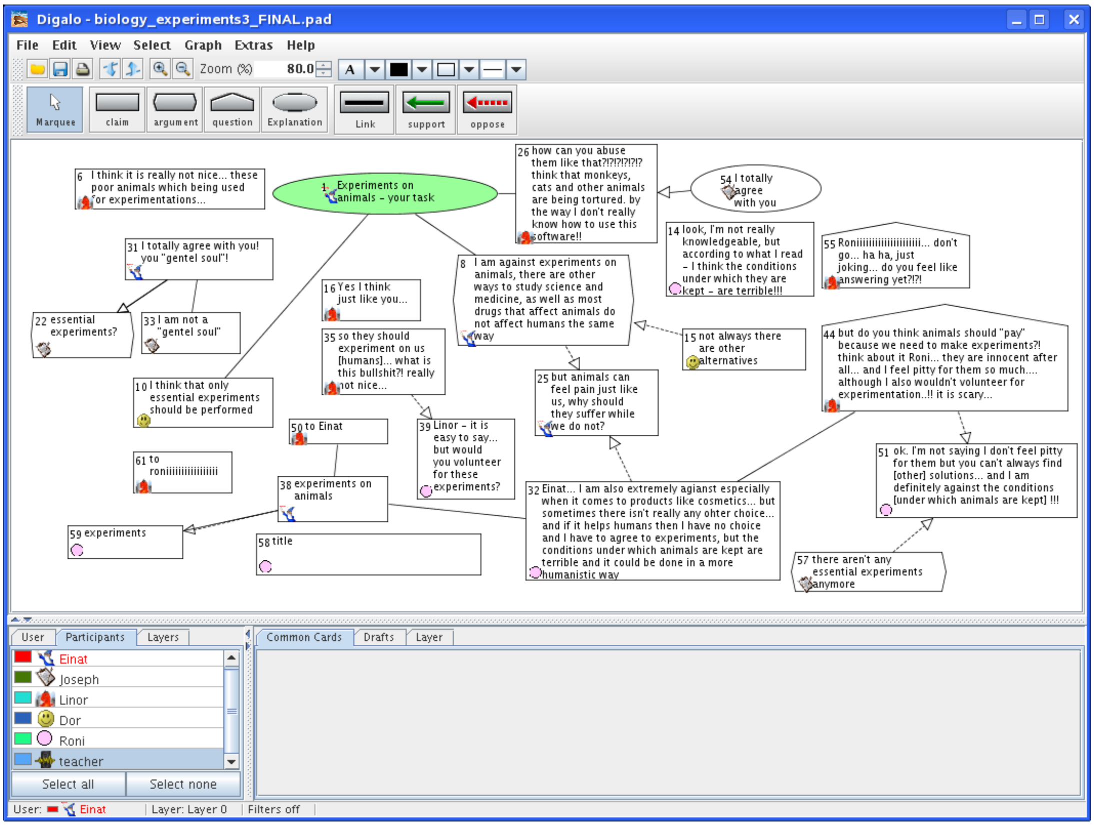
This chapter discusses how machine learning techniques have been utilized within the Argunaut project to build classifiers that can identify pedagogically important aspects in graphical e-discussions, such as off-topic contributions, reasoned claims, and question-answer interactions. While this chapter focuses on the machine learning approach within the Argunaut project, it should be noted that the induced classifiers have been integrated with the CASE engine, discussed in Chapter 5. This integration provides a proof-of-concept demonstrating how the CASE engine can be easily extended with external analysis modules. Similar to the approach described here, new machine learned classifiers, specifically targeted at driving the generation of automated feedback, may be developed and integrated with the CASE engine.

Section 6.1 introduces the Argunaut project, which constitutes the background of the line of research discussed in this chapter. Section 6.2 motivates and discusses, in greater detail, the specific research question tackled. Section 6.3 gives a general introduction to basic machine learning concepts and some major algorithms relevant to the presented work. Section 6.4 describes the methodology employed to build and validate classifiers. Section 6.5 reports on the empirical results obtained in validation experiments, which generally show the promise of the approach. Finally, Section 6.6 summarizes and discusses the results achieved also addressing important limitations.

6.1 Background: The Argunaut Project

The Argunaut project was aimed at building an e-moderation environment (Salmon 2004), enhanced with moderator-assistance technologies, for students learning argumentation. Students discuss and debate contentious topics by jointly creating argument diagrams, that is, new messages are "posted" as graphical boxes to a shared workspace; "replies" are indicated through graphical links.

Figure 36 shows an example e-discussion in Argunaut. Node and link types are represented through different graphical shapes (e.g., a rectangle, an oval, a diamond) and arrows (e.g., a solid green line, a dashed red line), which can be selected from a *palette* (see Figure 36, palette panel at the top of the screen). Argunaut utilizes an informal ontology (Schwarz and Glassner 2007), meaning that node and link types are based on typical moves in everyday conversations, and thus, can be understood and used without much theoretical background and training (node types: *claim*, *argument*, *question*, *explanation*; link types: *support*, *oppose*, *link*). The discussion is typically rooted in a controversial topic, which is given by the teacher in form of an initial node in the diagram (see Figure 36, green node with title *Experiments on animals—your task*, indicating the given ethical dilemma: Is it ethical to perform experiments on animals?).



1 Experiments on animals – your task

6 I think it is really not nice... these poor animals which being used for experimentations...

26 how can you abuse them like that?!?!?!?!?!? think that monkeys, cats and other animals are being tortured. by the way I don't really know how to use this software!!

54 I totally agree with you

31 I totally agree with you! you "gentel soul"!

8 I am against experiments on animals, there are other ways to study science and medicine, as well as most drugs that affect animals do not affect humans the same way

14 look, I'm not really knowledgeable, but according to what I read – I think the conditions under which they are kept – are terrible!!!

55 Roniiiiiiiiiiiiiiiiiiiii... don't go... ha ha, just joking... do you feel like answering yet?!?!

16 Yes I think just like you...

22 essential experiments?

33 I am not a "gentel soul"

35 so they should experiment on us [humans]... what is this bullshit?! really not nice...

15 not always there are other alternatives

44 but do you think animals should "pay" because we need to make experiments?! think about it Roni... they are innocent after all... and I feel pitty for them so much.... although I also wouldn't volunteer for experimentation..!! it is scary...

10 I think that only essential experiments should be performed

25 but animals can feel pain just like us, why should they suffer while we do not?

50 to Einat

39 Linor – it is easy to say... but would you volunteer for these experiments?

51 ok. I'm not saying I don't feel pitty for them but you can't always find [other] solutions... and I am definitely against the conditions [under which animals are kept] !!!

61 to roniiiiiiiiiiiiiiiiii

38 experiments on animals

32 Einat... I am also extremely agianst especially when it comes to products like cosmetics... but sometimes there isn't really any ohter choice... and if it helps humans then I have no choice and I have to agree to experiments, but the conditions under which animals are kept are terrible and it could be done in a more humanistic way

59 experiments

58 title

57 there aren't any essential experiments anymore

Figure 36: Example of a graphical discussion in Digalo, one of the tools supported within the Argunaut e-moderation environment. From McLaren et al. (2010).

In a typical classroom setting, one teacher-moderator monitors and supports multiple e-discussions (e.g., six groups with three to five students each). This process of monitoring and supporting multiple synchronous discussions in parallel is inherently difficult. The teacher-moderator must track and maintain a mental model of multiple discussion threads at a time. Important events in different discussions may occur in rapid succession, sometimes even in parallel. While monitoring or supporting one discussion thread, important events in other discussions might pass unnoticed. The Argunaut project investigated how the moderation process could be facilitated by means of a computer-based *Moderator's Interface* (see Figure 13, p. 122), which provides awareness indicators and alarms to highlight noteworthy situations, and feedback tools to intervene and remediate identified problem.

From a computational perspective, two different kinds of awareness indicators are provided in Argunaut. *Shallow indicators* are computed in relatively straightforward ways, e.g., through keyword search or descriptive statistics of node and link type usage. *Deep indicators* are more complex to compute but also potentially more meaningful to teachers. They are based on classifiers built using artificial intelligence techniques, in particular, machine learning, case-based reasoning, and natural language processing. This chapter focuses on experiments with machine learning techniques.

6.2 Research Question

To analyze discussion and argumentation activities in a systematic and scientifically sound way, researches in the CSCL community have developed different analytic frameworks (Clark et al. 2007). Such analytic frameworks typically consist of a number of categories, organized within different analysis dimensions, and detailed instructions how to assign these categories to discourse segments. The specific set of categories, as well as the definition of what exactly constitutes a segment, essentially depends on the specific research question to be investigated, the theoretical stance the researcher takes, and pragmatic considerations.

The pedagogical experts within the Argunaut consortium have developed such an analytic framework, which combines theoretical considerations (in particular, the notion of dialogism; Wegerif 2006) with practical concerns (insights from social learning and e-moderation research). In contrast to many other approaches, the Argunaut analytic framework focuses specifically on the analysis of graphical e-discussions. It distinguishes three levels of analysis, which can be associated with specific diagram structures:

- *node level:* individual contributions, e.g., off-topic contributions and reasoned claims
- *paired-node level:* one-step interactions (adjacency pairs), e.g., question-answer pairs, contribution-counterargument pairs

- *cluster level:* multi-step interactions, e.g., chains of opposition

Researchers can apply this analytic framework manually, e.g., to characterize graphical e-discussions in terms of how often certain behaviors occurred, and based on this, come to an overall assessment, an approach frequently used in CSCL research. However, the manual coding of e-discussions was in Argunaut only a first step towards the ultimate goal of automating the coding process. Once the coding process is automated, e-discussions can be analyzed automatically, and condensed to a few meaningful indicators, which provide human moderators with an intelligible overview of what is going on. So the crucial question is whether it is possible to automate the application of the Argunaut analytic framework through machine-learned classifiers.

> **(RQ4) Pattern Induction Mechanism:** "Can supervised machine learning techniques be successfully used to automatically induce computational models that identify important qualitative aspects in graphical e-discussions?"

With the first results published in 2007 (McLaren, Scheuer, et al. 2007), the work presented in this chapter directly builds upon pioneering work of Rosé and colleagues on automating the analysis of student discussions in CSCL settings (Dönmez et al. 2005; Rosé et al. 2008). Obviously, the presented approach differs from theirs in that different analytic frameworks have been automated: They used the Weinberger and Fischer (2007) framework while the Argunaut approach uses the framework sketched above. But there are also a number of other important differences. While their approach is based on threaded discussions, the Argunaut approach addresses graphical e-discussions, an instructional method nowadays well established in the CSCL community. In fact, the Argunaut approach is the first approach that addresses graphical e-discussions. Their discussions were manually pre-segmented into meaningful units of analysis before machine learning was applied. Consequently, the resultant machine-learned classifiers can only be applied to pre-segmented data. In contrast, the Argunaut approach capitalizes on the natural structuring of a diagram (individual nodes and linked node pairs) and can therefore be applied in a fully automated fashion, without any human intervention (or an additional technical segmentation component) required. In summary, the work presented in this chapter adds to the knowledge of how machine learning can help imbuing CSCL systems with diagnostic functions, which may be used to support moderators but could also inform the automated generation of adaptive support targeted at students.

6.3 Background: Machine Learning

Before the particular approach used in the Argunaut project is described, this section gives a short introduction to the basic concepts of machine learning (section 6.3.1) and its main algorithms (section 6.3.2).

6.3.1 Basic concepts of machine learning

The discipline of machine learning (Mitchell 1997; Witten and Frank 2005) investigates how computer algorithms can be used to automatically detect meaningful patterns in structured data representations. The description here focuses on flat data representations, meaning that data can be represented in a tabular format. Each table row represents one specific *example* (also called *instance* or *data points*). The typical assumption is that examples are statistically independent from one another. Each table column represents one specific *attribute* (also called *feature*). Each table cell defines the value of a specific attribute for a specific example. Overall, the table describes a set of examples in terms of predefined attributes, or, in mathematical terms, as *attribute vectors*. In *supervised machine learning* scenarios, one specific attribute is denoted as the target attribute. Supervised machine learning algorithms try to compute a model able to reliably predict the value for the target attribute based on regularities and patterns in the other attributes. If the target attributes takes *categorical* (or *nominal*) values, the problem is referred to as a *classification problem*. The target attribute is then typically referred to as the *class attribute*; its different values are referred to as *class labels* or simply *classes*. If the target attribute takes *numeric* values, the problem is referred to as a *regression problem*.

Table 31 shows an exemplary fictitious data set based on the “weather” data from Witten and Frank (2005). This data is used as a running example in the following descriptions. The table shows seven examples each representing a particular day described in terms of five attributes. Four attributes represent weather conditions on that day, in particular, outlook, temperature, humidity and whether it is windy or not. The fifth attribute, the target attribute *Play*, represents the decision whether to play some game on that day. The goal is to learn a model that can predict this decision based on the weather conditions of a given day. Since *Play* is a nominal attribute, the weather problem can be characterized as a classification problem. Some machine learning algorithms only work with nominal attributes. In such cases, numeric attributes may be transformed into nominal attributes through a process called binning. For instance, temperatures may be discretized using the bins *low*, *medium*, and *high*.

Table 31
Exemplary machine learning data set based on the "weather" data

Example ID	**Attributes**				**Target Attribute**
SeqNum	**Outlook**	**Temperature**	**Humidity**	**Windy**	**Play**
(nominal)	(nominal)	(numeric)	(numeric)	(nominal)	(nominal)
1	sunny	85	85	false	no
2	sunny	80	90	true	no
3	overcast	83	86	false	yes
4	rainy	70	96	false	yes
5	rainy	68	80	false	yes
6	rainy	65	70	true	no
7	overcast	64	65	true	yes

Note: Adapted from Witten and Frank (2005, p. 12).

Obviously, the ultimate goal is to build models that reliably predict the value of the target attribute for examples for which this value is not known in advance. In particular, the model should not incorporate particularities of the given example set not representative of the target population, a problem known as *over-fitting*. Therefore, the development of machine-learned models typically involves two phases: a *training phase*, in which a model is learned from a given set of examples, and a *test phase*, in which the model is applied to examples not used in the training phase to see whether the model generalizes beyond the training examples. If the model is over-fitted to the training set, its performance on an independent *test set*—and on the target population more generally—is expected to be limited. So if a set of labeled examples is given (that is, examples for which the value of the target attribute is known), one can split this set into a training set and an independent test set, learn a model from the training set, and check its predictive performance against the test set.

To make more economical use of a typically limited amount of available examples, an iterative procedure called *cross-validation* is often used. In a ten-fold cross-validation, the example set is split into ten subsets. In each iteration, nine subsets are used to train a model and the remaining subset to test the model. In each iteration, a different subset is used for testing. That is, across all iterations, each example is tested exactly once. The performance of the model is estimated by averaging across the performance scores of the individual iterations. A special case of cross-validation is *hold-one-out cross-validation*. That is, in every iteration, only one example is held out for testing purposes.

6.3.2 Machine learning algorithms

Many different supervised machine algorithms are available to induce models from data. Some of the most widespread and successful ones are available in standard machine learning toolkits, such as RapidMiner (Mierswa et al. 2006) and WEKA (Hall et al. 2009):

1R (Holte 1993): The 1R (one-rule) machine learning algorithm analyzes the training set and selects the one attribute that best predicts the target attribute. This extremely simple approach has shown to perform surprisingly well in practical applications, often only a few percentage points less accurate than far more complex approaches. For instance, when applied to the weather data set, 1R may select the *Outlook* attribute as the best predictor for the *Play* attribute, resulting in the rule depicted in Table 32.

Table 32
Exemplary 1R model based on the weather data

```
IF outlook = "sunny" THEN "no"
  ELSE IF outlook = "overcast" THEN "yes"
  ELSE IF outlook = "rainy" THEN "yes"
```

Note: Example based on Witten and Frank (2005, p. 85).

Decision tree learners (e.g., C4.5; Quinlan 1993): For the sake of simplicity the following description assumes a classification problem and nominal attributes. There are decision tree implementations that can also address regression problems and numeric attributes.

A decision tree is a hierarchical model. To classify an example, the example traverses the tree from the root to a leaf node. Each node within the tree represents a test on an attribute. Based on the test result, an edge is chosen to traverse to the next level of the tree (essentially each edge represents one attribute value). This procedure is repeated until a leaf node is reached, which represents the final classification.

Figure 37 shows an exemplary decision tree based on the weather data. For instance, an example with *Outlook* = *sunny* and *Humidity* = *high* would result in the prediction *no* for the *Play* attribute. This specific example would take the leftmost path through the tree.

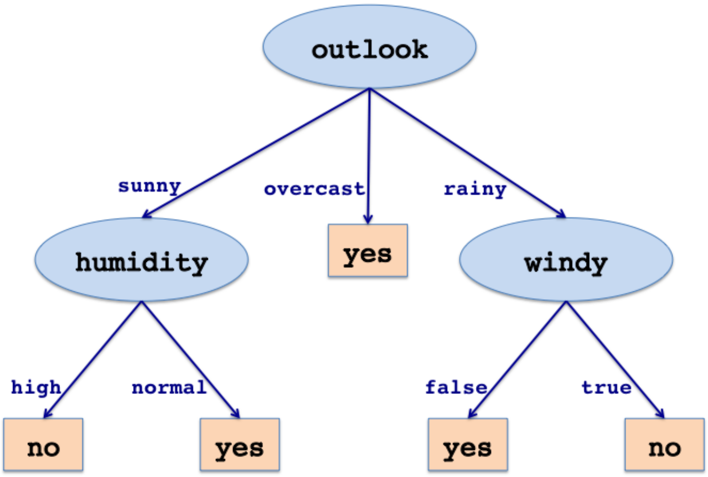


Figure 37: Exemplary decision tree model based on the weather data. Adapted from Witten and Frank (2005, p. 101).

Decision tree learners construct a decision tree from a given training set. First, the most informative attribute is selected and put at the root of the tree. The most informative attribute is the one that splits the training set into subsets as homogeneous as possible with respect to the class attribute. An optimal split, which is typically not possible, would cleanly separate the examples according to their class, that is, each subset would only contain examples with the same class label. Depending on the specific algorithm, different criteria may be used to determine the most informative attribute. For instance, C4.5 uses gain ratio as criterion, which is a normalized version of an information theoretical measure called information gain. After the root node has been created, its child nodes are created using the same procedure but only considering the training examples that would reach the node under consideration when traversing the tree (rather than the complete training set). The algorithm is repeated until none of the attributes would lead to a cleaner separation. The leaf nodes of the tree are then associated with the class that most frequently occurs within the subset of examples that reach the node. Typically, a procedure called pruning is used to remove branches in the tree that do not considerably improve the performance. In fact, such branches are often the result of over-fitting and are likely to reflect idiosyncrasies of the training set. That is, removing these branches is likely to improve the performance on unseen examples of the target population.

Naïve Bayes (McCallum and Nigam 1998): Naïve Bayes is a classification approach based on statistical modeling. It classifies examples by choosing the class label that is most probable given the values of the other attributes. It is called "naïve" because it uses the simplifying assumption that attributes are statistically independent from one another (given the class attribute). This assumption allows building a far less complex model, since parameters that represent the influence of attribute combinations on the class prediction are not included. Although this assumption is almost always violated, Naïve Bayes has shown to perform quite well in many practical applications. To learn a Naïve Bayes model from a training set, standard statistical parameter estimation procedures, such as the maximum likelihood method, can be used.

Table 33
Exemplary Naïve Bayes model based on the weather data and how it can be derived

$(\mathbf{1})\ P(Play \mid Outlook, Temp, Humid, Wind)$

→ **Apply Bayes' Theorem**

$$(\mathbf{2}) = \frac{P(Outlook, Temp, Humid, Wind \mid Play) \ast P(Play)}{P(Outlook, Temp, Humid, Wind)}$$

→ **Drop denominator since it is independent from the class attribute (takes the same value for each class)**

$(\mathbf{3}) \propto P(Outlook, Temp, Humid, Wind \mid Play) \ast P(Play)$

→ **Apply conditional independence assumption**

$(\mathbf{4}) = P(Outlook \mid Play) \ast P(Temp \mid Play) \ast P(Humid \mid Play) \ast P(Wind \mid Play) \ast P(Play)$

Table 33 shows an exemplary Naïve Bayes model based on the attributes in the weather data set (with numeric attributes transformed into nominal one). The presented formula allows assigning to each class label a value proportional to its probability, given all other attribute values. The final classification is the class label with the highest value (i.e., the most probable class label according to the model). In the concrete example, the goal is to find a model that predicts the value of the *Play* attribute given the *Outlook*, *Temperature*, *Humidity*, and *Windy* attributes, see line 1. The application of Bayes' Theorem yields the expression in line 2. The denominator does not depend on the target attribute *Play* and therefore takes the same value for each class label. Since it is a constant factor not influencing the ordering of class label probabilities, it can be discarded from the formula, yielding the expression in line 3. Finally, the conditional independence assumption is used, yielding the expression in line 4, which poses a far more tractable problem in terms of statistical parameter estimation. The parameters of the five probability functions can be easily estimated based on the frequencies in the training set.

Support Vector Machines (Boser et al. 1992): For the sake of simplicity the following description assumes a binary classification problem (i.e., positive and negative examples) and numeric attributes. There are Support Vector Machine implementations that can also address multi-class problems, regression problems, and nonnumeric attributes.

Support Vector Machines (SVMs) are based on a vector space model, that is, examples are thought of as vectors in a vector space. Each attribute defines one dimension of that vector space. A SVM defines a hyperplane to separate positive and negative examples in the vector space. To classify an example, it is mapped into the vector space. It will be classified as a positive or a negative example depending on the side of the hyperplane where it is located.

Figure 38 shows an exemplary SVM model based on the attributes *Humidity* and *Temperature* of the weather data set. Of course, SVMs can generally use an arbitrary number of attributes. Furthermore, in this specific example, positive and negative training examples can be linearly separated based on the attributes *Humidity* and *Temperature*.

To learn a SVM model, the separating hyperplane is computed in a way that maximizes the minimal distance of positive and negative examples to that hyperplane. The rationale is to maximize the margin that separates positive from negative examples, and thus to reduce the generalization error. For instance, in Figure 38, one can imagine an infinite number of possible separating hyperplanes. But the chosen one is optimal with respect to the width of the margin. Therefore, SVMs belong to the class of large-margin classifiers. Finding the hyperplane is a mathematical optimization problem—more specifically, a quadratic programming problem—for which efficient algorithm exist. Since it will not be possible to strictly separate positive from negative training examples in most cases, so-called slack variables are added to the equation system, which introduce penalty terms for training examples that are not compatible with the SVM model (i.e., training examples on the "wrong" side of the hyperplane).

Many classification problems are not linearly separable, that is, the boundary between positive and negative examples cannot be approximated well with a linear hyperplane. Therefore, SVMs make use of the Kernel trick, which essentially means to transform the original vector space into a higher-dimensional vector space through a kernel function. The SVM algorithm then determines a linear hyperplane in the higher-dimensional space, which is equivalent to a non-linear separation in the original space (e.g., a separation that can be described by a polynomial expression). In this respect, SVMs are more flexible compared to approaches such as Naïve Bayes, which can only represent linear classification functions, due to the assumption that attributes are conditionally independent. SVM induction is one of the most sophisticated and widely used supervise machine learning methods nowadays.

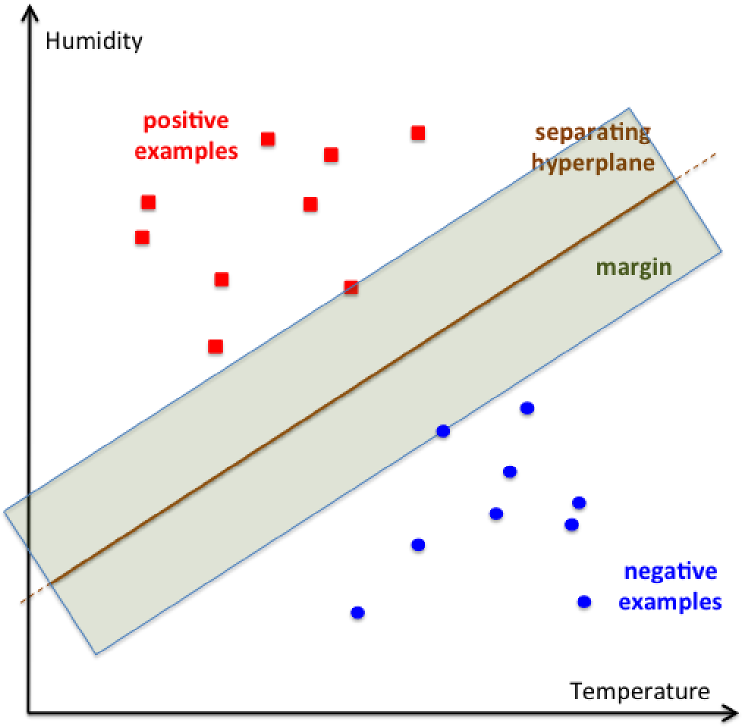


Figure 38: Exemplary Support Vector Machine model.

Boosting (e.g., AdaBoost; Freund and Schapire 1997): Boosting algorithms belong to the class of ensemble methods, that is, machine-learning algorithms that combine the predictions of multiple models (e.g., decision trees) to achieve an overall performance above the performances of each individual model. For instance, let's have a look at an AdaBoost learner that uses internally a decision tree learner to construct models. In general, AdaBoost can use any other machine-learning algorithm internally. In a first iteration, AdaBoost applies the decision tree learner to the original training set to induce a first model. Typically, models are not perfect and classify only a subset of the given training examples correctly. AdaBoost determines which training examples are misclassified and increases in the next iteration the weights of these examples, and, vice versa, decreases the weights of correctly classified examples. That is, the second-round decision tree learner will put more emphasis on previously misclassified examples, yielding a decision tree that potentially performs better on previously misclassified examples and potentially worse on previously correctly classified ones. The AdaBoost algorithm repeats this procedure over a predefined number of iterations to compute a new model in each. In addition, AdaBoost keeps track of the relative performance of each model and assigns a weight accordingly. That is, good models receive higher weights than bad models. When applying the overall AdaBoost model, the internal decision tree models are applied successively and their *votes* are combined, based on the assigned weights.

Originally, algorithms like AdaBoost have been used to boost the performance of *weak learning algorithms* (e.g., 1R) into a *strong learning algorithm*. Whether a learning algorithm is strong or weak essentially depends on its ability to adjust to characteristics of the training set. A 1R model, for instance, is not very flexible in this respect since it bases its decision only on a single attribute. Therefore, 1R is said to be a weak learning algorithm. It turned out that boosting methods can also improve the performance of other, already powerful learning algorithms, such as decision trees. It has been proven that the overall AdaBoost model performs at least as well as the best internal model on the training set (*training error*) if the internal models perform at least slightly better than chance. Yet, the more important question is whether the iterative optimization of AdaBoost may lead to over-fitting. That is, the AdaBoost model may not perform so well on previously unseen examples of the target population (*generalization error*). Freund and Schapire (1999) argue, based on theoretical and empirical grounds, that AdaBoost is not particularly susceptible to over-fitting. In fact, AdaBoost is in several respects comparable to SVMs. Both approaches can be described within the same margin-maximization framework.

Attribute selection: Besides the actual learning task, additional methods may be used in advance to preprocess the data. A common approach is to not provide the full set of attributes to the machine-learning algorithm, but to preselect the most informative subset of attributes, or, vice versa, filtering out attributes that may not contribute a lot. The reason is that many irrelevant or redundant attributes not only consume unnecessary computational resources during machine learning, but may also add noise, with potentially negative effects on the predictive performance. For instance, Kohavi and John (1997) showed that decision tree induction and Naïve Bayes-based modeling benefit from attribute selection in terms of improved prediction performance and smaller and therefore more comprehensible models. One attribute selection method is to use the correlation between attributes and the target attribute to determine potentially good predictors (Ng et al. 1997).

6.4 Machine Learning Methodology

The machine learning methodology in the Argunaut project essentially unfolded into two main stages. Section 6.4.1 describes how the data to which machine learning algorithms were applied was collected and prepared. Section 6.4.2 describes the general setup and approach to induce and validate classifiers.

6.4.1 Data collection

The goal in the Argunaut project was to build machine-learned classifiers on the node-level (i.e., classifiers that assign labels to individual nodes) and on the paired-node-level (i.e., classifiers that assign labels to pairs of nodes that are connected through a link). The classifiers should essentially emulate the behavior of human coders who apply the Argunaut analytic framework discussed above. As a first step, it was necessary to collect human-coded data, that is, nodes and node pairs labeled by human experts according to the Argunaut analytic framework. Supervised machine learning algorithms can then be used to associate certain properties of nodes and node pairs (e.g., language patterns in the contained text) with the labels assigned by human experts (i.e., in machine learning terms, the target categories).

Human experts analyzed a set of preexisting graphical e-discussions using the Argunaut analytic framework. For that purpose, it was necessary to generate coding sheets, in particular, Excel spreadsheets that allow human coders to conveniently assign labels to nodes and node pairs. Essentially, each table row corresponded to one unit of analysis (i.e., a node or a connected node pair). A first set of columns represented e-discussion data important for the coding process, e.g., the text contained in each node. A second set of columns represented the different coding dimensions of the analytic framework, one column per dimension. Human coders entered for each unit of analysis and coding dimension their assessment in the corresponding cell. For instance, on the node-level coding sheet, one column represented the coding dimension *off-topic*. Human coders entered for each node a *yes* or *no* into the corresponding cell depending on whether they judged the node to be an off-topic or on-topic contribution. To make this decision, they inspected the data in the coding sheet and checked, in addition, the original diagrams in the e-discussion tool, if necessary. To generate these Excel sheets, a software tool was developed to:

- import e-discussion data (XML log files) into a relational database,
- reconstruct the final state of the diagrams (the original data represented the trace of user actions rather than the final state of the diagram),
- extract the units of analysis (i.e., nodes and connected node pairs) and associate data important for the coding process with each unit of analysis, based on requirements defined by the human coders, and
- generate the Excel coding sheets.

Since the analytic framework was newly developed for the purposes of Argunaut, it was necessary to check the validity of the coding procedure, in particular, its inter-coder reliability. Inter-coder reliability indicates the level of agreement between multiple independent coders. A high level of agreement indicates objectivity (independence from the specific coder) and reliability (reproducibility of results), two important prerequisites for the validity of the analytic framework and approach. Therefore, multiple human coders coded the same subset of data independently to check the level of inter-coder agreement. The level of agreement was quantified through the Kappa (κ) statistic, in particular Cohen's κ (1960) when two coders did the coding, and Fleiss' κ (1971) when three coders did the coding. A κ value of 1.0 signifies perfect agreement, a κ value of 0 means agreement at chance level, and κ below 0 means agreement worse than chance. Krippendorff (1980) proposes to consider κ values above .7 as acceptably reliable in content analysis. Following this advice, further efforts were concentrated on the coding dimensions with κ values near or above .7.

Ultimately, five (out of seven) node-level dimensions, and four (out of five) paired-node-level dimensions surpassed the threshold and were kept for further analysis. The coding dimensions and results are reported in Table 34 (node-level) and Table 35 (paired-node level). Note that the number of examples varies for (in particular) different node-level categories. This is because there were several iterations of coding and machine learning in the course of the Argunaut project. In later iterations, efforts were concentrated on the most promising categories.

Table 34
Node-level categories and corpus statistics

Category	**Explanation / Coding**	**Examples**	**Positive instances**
Topic Focus	A node that focuses on the topic or task.	"Its not nice of human beings to exploit animals for their own needs. I think animals also have rights." (Counter-example) "I'm bored."	994 / 1188 84%
Reasoned Claim	An individual node that contains critical reasoning or argumentation (i.e., claim + backing). Student provides an explanation or some backing (e.g. evidence) to illustrate a position/opinion. If you can add "because" between two parts of the contribution, it is probably critical reasoning.	"I am against experiments on animals, because to my opinion it is not fair to use them against their will while they cannot reject." "Here it's not like with humans, as the father disengages from them, and he doesn't see them even in the afternoon, and he doesn't belong to the pack anymore"	500 / 1188 42%
Task Management	Comments about how to proceed with and manage the given task, such as "add titles," "write more," "answer him," etc.	"would you stop sending empty messages?!?!?!" "don't surf the net" "don't forget to add arrows"	98 / 968 10%
Request for Clarification	A request for clarification, reason, explanation, information, etc. from another person. Only applies when a contribution is "on topic."	"What are you basing this on?" "What do you mean by that?"	81 / 671 12%
Intertextuality	Explicit evidence of quoting or referring to external material. Only applies when a node is "on topic."	"It says in Wikipedia that …" "…in our discussions last week in class…"	23 / 671 3%

Note: Adapted from McLaren et al. (2010).

Table 35

Paired-node-level categories and corpus statistics

Category	Explanation / Coding	Example	Positive instances
Question-Answer	The 1st node is a question and the 2nd node is an answer to that question by a different student. Typically (but not necessarily) the type of link between the nodes would be "other."	1st node text: "In the wild does the father separate from the cubs or does he continue to live with them?" 2nd node, from a different person than the 1st shape: "They all live in a pack" (The link type between nodes is "other.")	117 / 775 15%
Contribution-Counter Argument	The 2nd node opposes the claim/argument raised in the 1st node and provides reasons or other type of backing for the opposing claim. Typically (but not necessarily) the type of link between the nodes is "opposition."	1st node text: "Do not separate, the male should be a partner in what happens even after the birth. The offspring is also his and he should take responsibility." 2nd node text, from a different person than the 1st node: "But in a situation like this the mother can get pregnant again and so might neglect a group of cubs." (The link type between nodes is "opposition")	224 / 768 29%
Contribution-Supporting Argument	The 2nd node supports the claim/argument raised in the previous one, and provides reasons or other type of backing for that claim. Typically the type of link between the nodes would be "support."	1st shape text: "We are against. Is it better for the male to get the female pregnant again so she'd abandon the babies?" 2nd shape text, from a different person than the 1st shape: "Separate. We think you should separate because you shouldn't hurt the mom, who will become a 'pregnancy machine' and move from one pregnancy to the next." (The link type between the nodes is "support.")	217 / 769 28%
Contribution-Question	The 2nd node is a question related to the 1st node. The links vary, based on the role of the question. If it's a rhetorical question, it may be an "opposition" link. If it's a genuine request for information etc., it will likely be "other."	1st node text: "She's also tormented because she's already exhausted as a result of all the pregnancies and also later on there's the risk that she'll neglect the cubs." 2nd node text, from a different person than the 1st node: "Are you for us or against us? Please answer our question." (The link type between the nodes is "other.")	100 / 776 13%

Note: Adapted from McLaren et al. (2010).

6.4.2 Experimental framework

The machine learning toolkit *RapidMiner* (Mierswa et al. 2006) was used to carry out machine learning experiments. RapidMiner provides a graphical user interface to define machine-learning workflows composed of a sequence of operators (e.g., to read data from a database or a file, to preprocess and transform data, to apply machine learning algorithms, to validate classifiers, etc.).

Since the human coders relied heavily on the natural language text, it was pretty obvious that an effective machine-learned model must do so as well. Therefore, it was necessary to bring the natural language texts into a form amenable to standard machine learning algorithms, more specifically, a representation in terms of attribute-value pairs. A common approach is a bag of words representation (Sebastiani 2002), that is, each word in a data corpus is mapped to one specific attribute. If the word appears in a text, the attribute is coded as a "1" and otherwise as a "0." Often, there are multiple variations (inflexions) of the same word to express specific grammatical categories, e.g., singular (*cat*) versus plural (*cats*) form. For the purpose of machine learning, only the basic meaning is typically of interest. Therefore, a process called stemming can be used to normalize all variations of a word to the same root form, e.g., the words *cat* and *cats* are presented by the same attribute. First, this reduces the overall amount of attributes (see the discussion of attribute selection in section 6.3.2). Second, without stemming, a machine-learning algorithm would not be able to recognize the semantic similarity of different inflexions of the same word, so stemming tells the machine learning algorithm to treat different inflexions of a word as the same semantic unit. The standard bag of words approach has one important limitation, namely, it cannot identify grammatical constructs that are represented in the sequence of words in a text. To approximate such sequential structures, one can consider pairs of consecutive words (*bigrams*) in addition to individual words (*unigrams*). On a more general level, the grammatical structure can be represented as the sequence of part-of-speech categories in the text. A common approach is to use a computational component called *part-of-speech tagger*, which assigns a part-of-speech category to each word, and then to approximate sequential dependencies by building bigrams based on consecutive part-of-speech categories (*part-of-speech bigrams* or, in short, *POS bigrams*). To experiment with different approaches to language analysis (e.g., turning on / off the generation of bigrams), the natural language processing toolkit *TagHelper* (Rosé et al. 2008) was used, which can analyze texts along the lines of what is described above. In particular, TagHelper was integrated with the RapidMiner toolkit as a new operator, which can be flexibly configured and combined with other machine-learning operators available in

RapidMiner (e.g., different preprocessing and learning algorithms). Besides language based attributes, the experiments included attributes that represent structural properties of the e-discussions (e.g., node type, number of inbound links to a node) and temporal properties (i.e., which node of a node pair was created first).

In earlier experiments (McLaren, Scheuer, et al. 2007; Scheuer and McLaren 2008), different machine learning algorithms and attribute sets were tested and compared. In the experiments reported here, the attribute sets of the best classifiers from earlier experiments were used in combination with algorithms that have been shown in the past to be effective for text categorization tasks, in particular, SVMs (Joachims 1998), Naïve Bayes (McCallum and Nigam 1998), and Boosted Decision Trees (Boosted DT) (Schapire and Singer 2000). Section 6.3.2 provides a brief introduction to the different methods. In addition, attribute selection was used, more specifically, χ^2-based attribute selection of the top 100 attributes. Since SVMs typically cope well with high dimensional input spaces, SVMs were also tested without attribute selection. Finally, SVM with cost balancing activated was tested, meaning that the relative weights (or misclassification costs) of the two classes were adapted to the class distribution during SVM training. In earlier experiments, this option led to an increased performance, presumably because of the skewed class distributions in the used data set (that is, positive examples largely outweighed negative examples or vice versa).

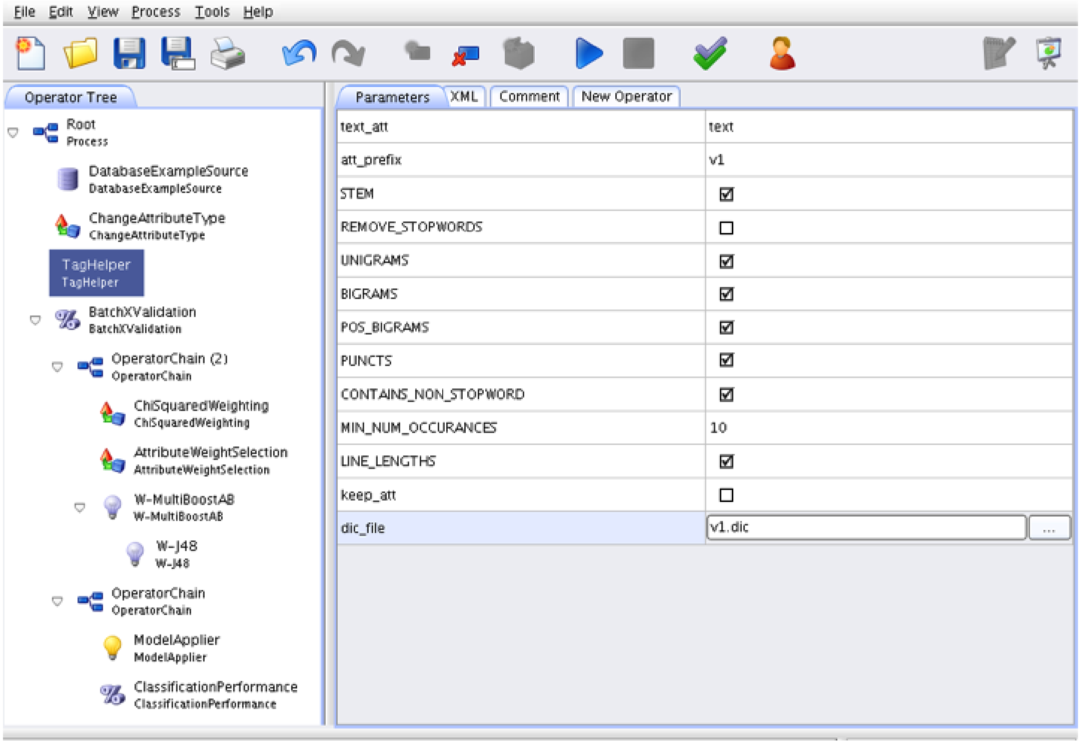
Classifier performance was measured using the Kappa (κ) statistic (Cohen 1960). In this case, κ measured the chance-corrected agreement between a machine-learned classifier and a gold standard (which is the human-assigned class). The κ statistic is not vulnerable to unbalanced class distributions and thus is a more appropriate criterion than the widely used error and hit rate (Ben-David 2006). The decision as to whether a classifier's performance is acceptable for real-world use depends on domain and application. An acceptability threshold of .8, or at least .7 (Krippendorff 1980; Rosé et al. 2008), is recommended in content analysis. Given that in Argunaut results are provided first to human teachers, who are aware of the possibility of possible misclassifications by the classifiers and the need to use their own judgment, a slightly more generous interpretation sufficient was considered. In particular, an acceptability threshold of .61 was adopted, which means, according to Landis and Koch (1977), an at least *substantial* agreement between a machine-learned classifier and a gold standard.

The performance of the classifiers was estimated by cross-validating data from one discussion (i.e., the test set) against the data from the remaining discussions (training set). Because data from one discussion was never in both the training and test sets simultaneously, statistical dependencies between training and test set were avoided, a problem often ignored and causing biased validation results.

Figure 39 shows a screenshot of an experimental setup used within the RapidMiner toolkit to build a classifier for the dimension *off-topic*. The panel on the left ("Operator Tree") displays the processing pipeline of the experiment. The process starts with retrieving relevant data from a database. This step is governed by the `DatabaseExampleSource` operator, which runs a predefined SQL query to retrieve positive and negative examples described in terms of relevant attributes from the database. (The next operator, the `ChangeAttributeType` operator, is not relevant to the discussion here.)

As mentioned above, the natural language processing toolkit TagHelper was integrated within RapidMiner. The corresponding TagHelper operator processes one particular attribute that represents the text of e-discussion contributions as an unstructured string (*text*). How TagHelper operates is governed by a number of options, displayed in the panel on the right in Figure 39. For instance, in the shown configuration, TagHelper uses stemming but no stopword filter, extracts unigrams, bigrams and part-of-speech bigrams, and generates attributes that indicate the overall text length and whether specific punctuation marks are contained in the text. TagHelper only considers words that occur at least ten times in the overall data set. Less frequent words are unlikely to be helpful for the classification task or may even cause noise. After generating the new attributes according to the just described specification, TagHelper discards the *text* attribute, which is of no use anymore.

The `BatchXValidation` operator contains two inner `OperationChains`, which specify nested processes. The first operator specifies the process of building a machine-learned classifier from a training set. The second operator specifies the process of checking this classifier against a test set. The `BatchXValidation` operator partitions the incoming data set into *n* data batches, each containing the data of precisely one e-discussion. The `BatchXValidation` operator then iteratively builds and evaluates *n* machine-learned model, using a different e-discussion in each iteration as the test set, and the remaining *n*-1 e-discussions as the training set. In this specific instance, the machine-learning training process comprises three steps: (1) determining for each attribute a weight factor based on the χ^2 statistic (`ChiSquaredWeighting` operator), (2) removing all but the 100 best attributes (`AttributeWeightSelector` operator), and (3) applying a machine-learning algorithm (`W-MultiBoostAB` operator). The boosting operator itself uses internally another machine learning algorithm, which is, in this specific case, a decision tree learner (`W-J48` operator). The performance is evaluated in each iteration by first applying the model to the test set (`ModelApplier` operator) and then checking which test examples are correctly and which ones incorrectly classified (`ClassificationPerformance` operator). The results of each iteration are condensed into an overall performance score, in this case, the κ statistic. The `BatchXValidation` operator aggregates the performance scores across all iterations into an overall performance estimate.



Root
Process
DatabaseExampleSource
DatabaseExampleSource
ChangeAttributeType
ChangeAttributeType
TagHelper
TagHelper
BatchXValidation
BatchXValidation
OperatorChain (2)
OperatorChain
ChiSquaredWeighting
ChiSquaredWeighting
AttributeWeightSelection
AttributeWeightSelection
W-MultiBoostAB
W-MultiBoostAB
W-J48
W-J48
OperatorChain
OperatorChain
ModelApplier
ModelApplier
ClassificationPerformance
ClassificationPerformance

text_att	text
att_prefix	v1
STEM	☑
REMOVE_STOPWORDS	☐
UNIGRAMS	☑
BIGRAMS	☑
POS_BIGRAMS	☑
PUNCTS	☑
CONTAINS_NON_STOPWORD	☑
MIN_NUM_OCCURANCES	10
LINE_LENGTHS	☑
keep_att	☐
dic_file	v1.dic ...

Figure 39: Setup of machine learning experiment in RapidMiner.

Essentially, similar experimental workflows for different machine-learning approaches were defined (i.e., different algorithms, attribute selection turned on and off) to determine which approach works best and at which performance level. The best classifiers that surpassed the a priori defined performance threshold were integrated in an online web service (*Deep Loop*), which could be invoked by teacher-moderators through the Moderator's Interface. This web service was also integrated with the CASE engine to make machine-learned classifiers accessible in the LASAD system.

6.5 Machine Learning Experimentation

This section discusses the results achieved for classifiers on two levels of analysis: node-level classifiers (section 6.5.1) and paired-node-level classifiers (section 6.5.2).

6.5.1 Node-level classifiers

The experimentation used a data-centric approach by encoding as much information as possible in attribute-value form, without considering the specific categories of interest, in hopes that the inference mechanism itself would focus on and use the most predictive attributes. Nodes were analyzed in terms of structural attributes (node and link types, incoming and outgoing links) and textual attributes (textual content of nodes extracted by TagHelper). Some attributes were dropped (e.g., number of in-links of type *opposition*) when initial machine learning experiments indicated they did not improve the results. The full set of attributes finally used for node-level machine learning is shown in Table 36.

Table 36
Attributes used in node-level machine learning experiments

Type of attribute	Specific attributes
Structural	Node type Number of undirected links Number of in-links Number of out-links
Textual (derived using TagHelper)	Unigrams: simple terms (equivalent to keyword search) Bigrams: consecutive terms (paired word phrases, such as ‘common denominator’) POS bigrams: part-of-speech bigrams (shallow syntactical structures, e.g. Noun-Verb, Adjective-Noun) Punctuation: Obviously, a question mark is a strong indicator of a question. Text Length: The overall text length of the contribution.

Note: Adapted from McLaren et al. (2010).

Originally, all of the categories shown in Table 34, p. 204, were targeted, but after initial machine learning experiments, both annotation efforts and machine learning experiments were concentrated on only two categories, *Topic Focus* and *Reasoned Claim*. The other three categories (i.e., *Task Management*, *Request for Clarification*, *Intertextuality*) led to weaker machine learning results in earlier experiments (i.e., *κ* values well under .60) most likely because of imbalanced class distributions and too few examples for one class, two problems well known for their detrimental effects on machine learning (Japkowicz and Stephen 2002; Weiss 2004). All but one category, *Reasoned Claim* with a proportion of positive instances of 42%, showed an overwhelming majority of one class, with proportions ranging between 84% (the positive *Topic Focus* annotations) and more than 97% (the negative *Intertextuality* annotations). The lack of success in automated learning of some of the discarded categories may also be attributed to the "ill-definedness" of those categories; more specifically, while humans were able to consistently identify members of the categories, the key attributes of the categories may be too difficult for a computational approach to identify and/or use.

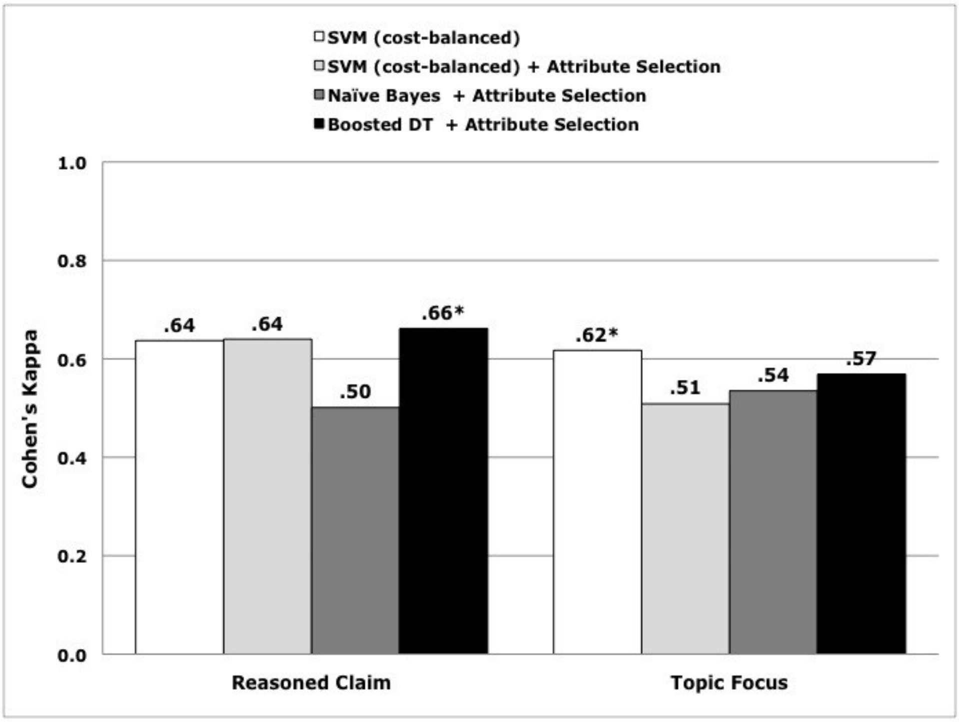


Figure 40: Results obtained in node-level machine learning experiments. From McLaren et al. (2010).

Figure 40 shows the results at the node level. Performances well above chance were achieved with all algorithms. Four of five *Reasoned Claim* classifiers surpassed the acceptance threshold of $\kappa > .61$. The best result was achieved using Boosted Decision Trees combined with attribute selection ($\kappa = .66$; 83.6%). Results for *Topic Focus* were somewhat lower: only the SVM classifier without attribute selection yielded acceptable results ($\kappa = .62$; 88.9%).

6.5.2 Paired-node-level classifiers

The specific attributes used for the paired-node experimentation are shown in Table 37. As with the node level, some attributes were dropped after initial machine learning experiments demonstrated that they did not improve results (e.g., the attribute representing the time elapsed between creation and first modification of the two involved shapes). Notice two key differences between this set of attributes and those of the node level (Table 36, p. 210). First, there are simply more attributes at the paired-node level. This is due to the greater structure involved at this level (i.e., the additional node, at least one link between the nodes, and the links associated with the additional node), as well as the additional text (i.e., two textual contributions instead of just one). Second, the paired-node level introduces the notion of temporal sequence. One node must have been created before the other, which is represented in the attribute data by the earlier node being designated *Node 1*, the later node as *Node 2*. The temporal sequence often also implies interaction between students; whether different students created the two nodes is also captured as an attribute. Note, however, that any participant in a discussion can create the link between two nodes, meaning therefore that connected nodes do not necessarily imply interaction between students. In practice, however, the second student almost always creates the link to the first student's node.

Table 37
Attributes used in paired-node-level machine learning experiments

Type of attribute	Specific attributes
Structural	Link type Both nodes created by the same user?
Structural + Temporal Sequence	Link and node 1 [node 2] from same user? Shape type of node 1 [node 2] Link direction (undirected, from node 1 to node 2, from node 2 to node 1)
Textual	Combined text length of node 1 and node 2 Difference in text length between node 1 and node 2
Textual (TagHelper) + Temporal Sequence	*Same as for the node level, as shown in Table 36 (i.e., unigrams, bigrams, POS bigrams, punctuation, text length), except applied both to node 1 and node 2 individually. Note that temporal sequence is implicitly introduced, since node 1, as well as all of its attributes, was created before node 2, and this ordering is then instantiated via the attribute names (e.g., node1_Textlength, node2_Textlength, etc.)*

Note: Adapted from McLaren et al. (2010).

As can be seen in Table 35, p. 205, all of the paired-node categories showed a substantial majority of one class, with proportions consistently larger than 71% (negative *Contribution-CounterArgument* annotations). Unlike the node level, all paired-node categories (i.e., *Question-Answer*, *Contribution-CounterArgument*, *Contribution-SupportingArgument*, *Contribution-FollowedByQuestion*) led to very promising early machine learning results and, thus, further efforts were concentrated on these categories.

For the paired-node experiments, the same experimental setup was used as for node-level classifiers, again experimenting with algorithms that have performed well in past text classification tasks. As can be seen in Figure 41, all machine learning algorithms (except SVM without attribute selection) yielded (close to) acceptable results (i.e., κ > .59). Boosted Decision Trees proved to be the most effective machine-learning algorithms for two of the four best paired-node categories (*Contribution-CounterArgument* and *Contribution-SupportingArgument*) with κ values of .71 (88.5%) and .66 (86.7%), respectively. In a third category (*Question-Answer*) Boosted Decision Trees and Decision Lists achieved a practically identical performance of κ = .78 (94.5% [Boosted Decision Trees] and 94.3% [Decision List]), outperforming the other three algorithms. In the fourth category (*Contribution-FollowedByQuestion*) SVM with attribute selection performed best, reaching κ = .75 (94.2%).

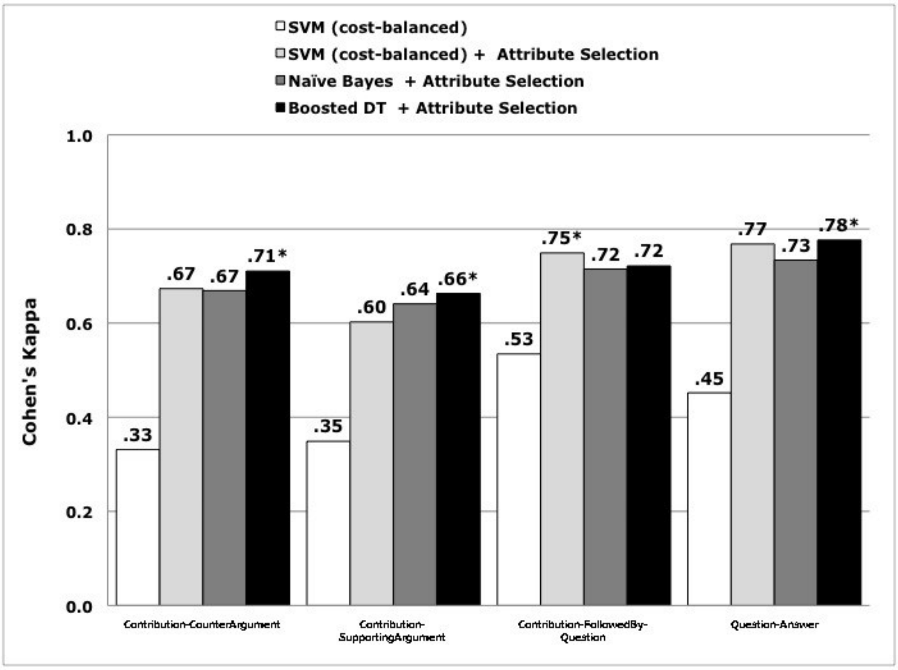


Figure 41: Results obtained in paired-node-level machine learning experiments. From McLaren et al. (2010).

6.6 Discussion and Conclusion

The results reported in this chapter show that machine learning is a promising approach to develop computational components able to automatically analyze important aspects of argumentation learning, or collaborative learning more generally. The presented approach is the first approach to analyze graphical pedagogical discussions using supervised machine learning. Empirical validation results indicated that the induced classifiers successfully identify qualitative aspects in e-discussions on two different levels—individual and paired contributions—based on a combined analysis of structural, temporal, and textual properties, despite the difficulty of an in part largely imbalanced class distribution, which is a known problem in machine learning research (Japkowicz and Stephen 2002; Weiss 2004).

The total number of examples used for machine learning was not really small in absolute numbers (more than 1000 node-level and 700 pair-level examples), but for text categorization tasks, much larger document sets are often used to capture variety and resolve the ambiguities in natural language use (Sebastiani 2002). Yet, others were successful using similar numbers of examples as was done here for similar tasks. For example, Dönmez et al. (2005) used approximately 1250 instances, Ai et al. (2010) about 700 instances to induce classifiers for qualitative aspects of argumentative educational discussions. So the size of the corpus seems to be sufficient although more empirical evidence is needed to arrive at a more definite conclusion on that point (e.g., some variability in the number of instances can be expected depending on the specific analysis categories, discussion topic, student population, and other circumstantial conditions). Moreover, the Argunaut approach could capitalize on temporal and structural aspects of graphical e-discussions, which provide additional cues for the classification task. Interestingly, the paired-node level classifiers were, overall, more accurate than the node level classifiers, potentially because the paired-node level provides more structure that machine learning algorithms can exploit.

In contrast to previous approaches (e.g., Rosé et al. 2008), Argunaut classifiers can operate in fully automated fashion, capitalizing on the pre-structuring inherent to discussion graphs. The approach builds upon state of the art machine learning technology, which is nowadays available in general-purpose toolkits, such as the here used RapidMiner (Mierswa et al. 2006), which is highly relevant from a practical perspective. The Argunaut analytic framework should be understood as just one example application. It was devised without taking any machine learning specifics into consideration to make the machine learning more feasible. Rather, the approach is strictly oriented towards educational theory and practice. So there is good reason to believe that the machine learning approach easily transfers to other analytic frameworks targeted at discussion and argumentation activities.

A notable restriction is that the Argunaut approach is limited to fixed structures, in particular, nodes and node pairs. The number of nodes and connected node pairs is typically relatively small in an e-discussion. Only in rare cases are there more than 30 nodes and links (a link essentially defines a node pair). This makes it possible to enumerate all nodes and node pairs and apply classifiers in an exhaustive way. Of course, this is not possible anymore if one considers arbitrary clusters, such as chains of opposition of an arbitrary length (i.e., the cluster level of the Argunaut analytic framework). The reason is that the number of possible clusters is exponential in the number of nodes. So even in relatively small graphs, it is problematic to analyze all possible clusters due to limited computational resources. Moreover, the definition of attributes is more challenging on the cluster level, since one cannot rely on a given fixed structure (e.g., a first node and a second node of a node pair). Rather, one would have to base the attribute definitions on invariant cluster properties, that is, properties that can be defined in each cluster independent of the specific node and link configuration, such as the concatenated node texts or the majority node type within a cluster. Yet, such invariant aggregate attributes cannot represent the internal structure of a cluster, e.g., which specific node contains which piece of text. But exactly these structures might be the defining characteristics of a given cluster type. In summary, the machine learning approach used for nodes and node pairs cannot be easily transferred to the cluster level.

In the context of the Argunaut project, my colleague Jan Mikšátko worked on an alternative approach, which uses heuristic search and graph-matching techniques to identify clusters similar to one or a small set of given model clusters that represent categories of the Argunaut analytic framework (Mikšátko 2007). Yet, this approach has its own shortcomings. First, the quality of the generalization is expected to be limited compared to machine learning, since only a few examples are considered rather than a representative set of examples. Second, the algorithm is designed as an information retrieval system that returns a ranked list of the best matches to the given model clusters. Currently, there is no way to decide how many of the results (if any) are really relevant with respect to the target category. So the algorithm is currently configured to always return the top five matches, no matter whether there are actually more or less relevant clusters in an e-discussion.

An important caveat in the presented approach is that the threshold of acceptance (κ = .61) is somewhat arbitrarily set. The only way to decide whether a classifier is good enough for practical use *is* practical use. Unfortunately, the Moderator's Interface, including the machine learned classifiers, was not ready for practical application before the final phase of the Argunaut project, leaving little opportunity to test the classifiers with real moderators in real classrooms. However, first results, anecdotal in nature, suggest that the classifiers may, in fact, help moderators to work more effectively and efficiently. In a study with one teacher, this teacher needed less time finding answers to her questions when the classifiers were available in the Moderator's Interface (Asterhan et al. 2008). Of course, to arrive at more definite conclusions, practical use on a larger scale in more rigorously controlled setting would be required. A different question is whether the performance is good enough to inform the automated generation of support for students without an intermediary, filtering moderator.

Another observation is that the classifier performances decreased somewhat from the results achieved with earlier classifier versions—classifiers trained with only a subset of the final data corpus (McLaren, Scheuer et al. 2007). This is somewhat counterintuitive, since one would expect that classifiers improve in performance when more training data is available. In fact, the very definition of machine learning, as provided by Tom Mitchell (1997, p. 2; italics mine), is along these lines:

> "A computer program is said to *learn* from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, *improves with experience E*."

However, the crucial point is that the "class of tasks T" may have changed by extending the set of examples. More specifically, data was collected in different countries (U.K., the Netherlands, and Israel), from different student populations (high school and University students), who discussed different topics (e.g., experiments on animals, or, the effect of ICT on learning experiences). So, by adding new data, the nature of the data corpus that the classifiers were trained on, and tested against, considerably changed. For instance, to analyze the off-topic category, not only topic-independent cues may play a role (e.g., an empty node is trivially off-topic), but topic-specific terms are presumably important as well. Moreover, the discussion culture in different countries and the age of discussants might have a big influence on language use. For instance, Schler et al. (2006) found that the age and gender of bloggers have significant impact on their writing style and preferred content. That is, a classifier trained with one student population might drop in performance when applied to other student populations that use language in a different way. In summary, by adding new data of a slightly different nature, the classification task potentially became more demanding for the classifiers, since they were expected now to analyze more heterogeneous data and patterns.

The more general lesson is that classifiers are typically not universally applicable, but restricted to scenarios that are sufficiently similar to the training scenario. This has important consequences with respect to the costs and efforts of building intelligent analysis functionality based on machine learning. For some time past, the natural language processing community is investigating how existing classifiers can be transferred, at low cost, from their original domain to related new domains (Daumé III and Marcu 2006). Recently, CSCL research started to address the problem of context-dependent classification as well (Mu et al. 2012).

Part C

Research Synthesis

Chapter 7
Summary of Results

In the preceding chapters, results regarding the overarching research question "How to Design Adaptive Argumentation Learning Systems?" were discussed. Each chapter contributes to important aspect of argumentation system design, in particular, the user interface, architectures for automated analysis and adaptive support, and specific approaches to automatically analyze argumentation-learning activities. Building upon the current state of the art, more specific research questions for each partial aspect have been formulated and researched. Table 38 and Table 39 give an overview of the specific research questions, the approaches taken in this dissertation to address these research questions, and the results obtained.

The first research question, addressed in Chapter 4, was concerned with the design of the graphical user interface of argumentation-learning systems. In particular, the question was whether the quality of student discussions could be improved by combining different approaches to structuring student learning activities. The two specific structuring elements used were argument diagrams (a knowledge representation approach) and sentence openers (a discussion scripting approach). The assumption that both approaches synergistically complement one another is based on the observation that each provides unique support on one specific level, namely the content level (diagrams support learners in better understanding the structure of and relations between arguments) and the social level (sentence openers support learners in engaging in fruitful argumentative discussions with others). Overall, skills on both levels must be combined to discuss a topic in a competent manner. Based on results from CSCL research, the FACT-2 collaboration script was developed, which defines, in a precise way, a learning process in which students analyze and discuss conflicting texts, supported through argument diagrams and a discussion script. An empirical study indeed showed that a full version of FACT-2, which involved both, diagrams and a discussion script, was superior to an ablated version of FACT-2, which only involved the diagrams but no discussion script. So the results suggest that students profit when argument-centered discussions are supported through a discussion script. Whether both levels of support reinforce one

Table 38

Summary of research questions of the dissertation and obtained results (RQ1 and RQ2)

Research question	Approach	Results
(RQ1) Multilevel Scaffolding: "Does a user interface that integrates argument diagramming with a discussion script promote the quality of student-to-student interaction and content learning more than each individual method?"	**FACT-2 script**: – Student pairs analyze and discuss opposing texts – Analysis supported through argument diagrams – Discussions supported through role script (proponent and critic) and corresponding sentence openers – Student activities organized into a sequence of four activity phases to guide students through the process	FACT-2 empirically tested against ablated script version without discussion script (i.e., roles and sentence openers): – Higher-quality discussions in terms of elaboration moves – Students assessed their learning of argumentation more positively **Bottom line:** Evidence indicating that students profit from scaffolding on multiple levels; open question regarding synergistic effects
(RQ2) Adaptive Support Architecture: "How can a software architecture be developed to optimally provide adaptive support across different argumentation domains and learning scenarios?"	**CASE engine (overall design):** – Software component that connects to argumentation learning systems over the network to deploy feedback agents (loose coupling) – Behavior of feedback agents highly configurable on three levels (patterns, messages, strategies) – Technical extension points to add new functionality programmatically – Graphical user interface to define and administer feedback agents online and in simplified fashion	Demonstration of generality through application in four different applications with different requirements: – LARGO: support in analyzing legal argumentation protocols – Science-Intro: support in mapping out the structure of scientific papers – Metafora: support in collaboratively deliberating about math and science problems in a graphical discussion space – Argunaut: support in moderating graphical e-discussions about controversial ethical topics Small-scale evaluation study indicates promising results with respect to the graphical feedback authoring and administration front-end **Bottom line:** Proof-of-concept of a technical design and user interface to easily customize adaptive support according to domain- / scenario-specific requirements

Table 39

Summary of research questions of the dissertation and obtained results (RQ3 and RQ4)

Research question	Approach	Results
(RQ3) Pattern Definition Mechanism: "How can a pattern search component be developed to optimally support the definition of patterns across different argumentation domains and learning scenarios?"	**CASE engine (pattern mechanism):** – Utilizes a rule-based engine to model diagrams and search for patterns – Data model allows the definition of domain patterns and collaboration patterns – Support of different pattern languages (Jess and XML) – Graphical language to specify patterns in a graphical user interface	Demonstration of generality through application in four different showcase applications with different requirements (see RQ2 – Results) Small-scale evaluation study indicates mixed results with respect to the ease of use of the graphical pattern language (most difficult aspects of user interface), suggesting that especially technical novices may need more extensive training, e.g., in concepts of logics **Bottom line:** Proof-of-concept of a configuration mechanism and graphical language to flexibly define patterns, which does reduce but not eliminate the complexities inherent to the concept of patterns
(RQ4) Pattern Induction Mechanism: "Can supervised machine learning techniques be successfully used to automatically induce computational models that identify important qualitative aspects in graphical e-discussions?"	**Argunaut machine learning approach:** – Two levels of analysis: nodes and node pairs – Creating a data corpus by hand-coding of graphical e-discussions according to a newly developed analytic framework – Representing examples in terms of linguistic, structural and temporal properties – Machine-learning experimentation with state of the art algorithm that have been successfully used in the past to analyze verbal data – Selection and deployment of classifiers that surpassed a given performance criterion	Validation results show that approach is feasible, even with an only moderately large data corpus: 2 node-level and 4 paired-node-level classifiers surpassed the performance threshold (but addition tests are needed to check whether threshold is appropriate) By design, classifiers can be used fully automatically in practical settings, that is, no additional human or automated segmentation required Identification of limitations with respect to analyzed structures (restricted to fixed-size structures) and classifier generality (performance loss when training and application contexts differ) **Bottom line:** First approach in CSCL to induce classifiers from graphical e-discussions data to automate the application of an analytic framework

another (i.e., operate synergistically) is up to future research to explore. Some first anecdotal evidence points in this direction. Despite the fact that FACT-2 proved to be successful compared to a somewhat simpler approach, there is one major drawback that FACT-2 shares with the vast majority of other CSCL scripting approaches, namely its static nature. That is, all students undergo exactly the same procedure, independent of their individual knowledge and skill level; the overall quality of their discussion, collaboration, and problem solving; their progress; and specific problems they encounter. Static approaches are not able to respond to individual differences, nor do they account for the dynamics of collaboration and learning processes. It has been noted that the very idea of providing a scaffold also eventually entails fading of the scaffold in order to gradually transfer competencies to increasingly independent learners (Pea 2004). Others pointed out that provided structures may get in the way of learning and interfere with already existing internal (mental) scripts (Dillenbourg 2002; Fischer et al. 2013). All in all, it would be desirable to adapt support to the actual needs, or to provide support only on a by-need basis. The topic of adaptation and automated analysis was addressed with the other three research questions.

A first technical challenge in imbuing argumentation-learning systems with automated support is how to design a software architecture for adaptive support (research question 2 addressed in Chapter 5). A number of (even if not many) adaptive argumentation systems exist and all of these systems must implement some technical solution to providing adaptive support. Rather than developing just another solution in fulfillment of particular objectives of a particular scenario, the second research question was how to design a more general software architecture, or software framework, that can be flexibly used to provide support across a whole range of argumentation domains and learning scenarios. The approach taken was to review what is implemented in existing systems, to design a general software framework that is principally flexible enough to mimic the functioning of the given systems, and to devise a configuration framework that allows customizing the system behavior according to specific requirements. The result of this effort is the CASE engine, a highly configurable and extensible software component to provide adaptive support for argument diagramming activities. The CASE engine allows defining feedback agents in terms of patterns, messages, and strategies, and deploying these feedback agents to learning sessions. Four showcase applications demonstrated the breadth of the CASE engine, addressing different argumentation domains (the Law, science, group deliberation, ethical debates), involving different learning activities (argument analysis, argument planning, argumentative discussions), supporting different learning objectives (domain-specific argumentation structures, collaboration processes), and employing different approaches to automated analysis (expert-defined patterns, machine-learned classifiers). Finally, a graphical user interface that allows defining and controlling feedback agents online and in simplified fashion was developed. A small-scale evaluation study showed that users generally appreciated the CASE Feedback Authoring Tool, found it well designed and easy to use. The CASE engine, as the first system of this kind, provides a proof of concept for providing adaptive support for argumentation-diagramming activities across scenarios and domains.

Along with the development of the CASE engine, another research question related to the previous one was addressed in Chapter 5. In particular, to adaptively support argument diagramming, a system must be able to analyze the student-created diagrams in order to diagnose possible problems or to generate hints regarding possible next steps. In accordance with the objectives of the overall CASE engine, an approach was needed to flexibly define patterns in diagrams across different domains and scenarios. To continually model the current state of diagrams and to apply pattern-identifying analysis rules to spot noteworthy situations, a rule engine—more specifically, the Jess engine (Friedman-Hill 2003)—was employed. This mechanism to identify patterns was integrated with the overall CASE configuration framework, that is, it was now possible to specify declarative pattern definitions in XML files and make use of these patterns in feedback agents to generate feedback messages. The four showcase applications discussed above also demonstrated the generality of the pattern definition approach. To make the pattern definitions accessible to technical novices, who are typically not able or willing to program production rules, a graphical language was devised and integrated within the graphical user interface of the feedback-authoring tool. This graphical language allows users to conveniently define the structure of patterns by creating LASAD diagrams; the resultant pattern specifications are automatically translated into operational production rules. This novel approach—I am not aware of any equivalent approach—was also empirically evaluated. It turned out, not surprisingly, that the definition of patterns, even if supported through a graphical language, poses a stiff challenge to users, particularly when they are not formally trained in logic and programming. Nevertheless, the results were overall encouraging, given that study participants did not have much opportunity to familiarize themselves with the CASE Feedback Authoring Tool and learn how to use it. More research is clearly needed to find ways to further support and facilitate this demanding task.

Finally, many patterns of interest cannot easily be defined by hand. For instance, some patterns are too complex, while for other patterns humans simply do not know how to describe them in a formal, computer-understandable way. Rather than taking a top-down, knowledge engineering approach, a viable alternative may be to take a bottom-up, data-driven approach. In particular, machine learning can be used to induce patterns automatically from data. So, the final research question, addressed in Chapter 6, was how feasible such an approach is, and, more specifically, whether it is possible to automate an existing analytic framework for graphical e-discussions through supervised machine learning. Within the Argunaut project, pedagogical experts coded e-discussion data using a multi-dimensional analytic framework specifically devised for that purpose. The resultant data corpus was used as input to machine learning experiments with a range of different attribute sets, machine learning algorithms, and parameter settings. The experiments addressed the node level and paired-node level. The attribute space was defined in terms of linguistic, structural, and temporal properties. The validation results showed the high potentials of a machine learning approach to analyze graphical e-discussions. Two node-level and four paired-node level classifiers, induced from a moderately sized data corpus, yielded satisfactory or better results. So the research question can be answered affirmatively. The approach provides a proof of concept and a potential model for future approaches to the automation of analytic frameworks for argumentation. Also two important limitations pointing to potential areas of future investigation were discussed: First, the approach is restricted to predefined, fixed-size structures (nodes and node pairs). Second, classifiers may drop in performance when applied in contexts different to the training context.

Chapter 8

Computer-Supported Argumentation: Theoretical and Practical Aspects

This chapter puts the results of this dissertation into a larger context. Potential benefits and limitations of different design options for the implementation of argumentation learning systems are summarized and discussed. Both, theoretical considerations and empirical results—obtained from this dissertation and previous research—are addressed. The discussion will give consideration to educational aspects (i.e., effects on learning processes and outcomes), computer science aspects (i.e., techniques for the computational analysis and support of learning processes), and practical aspects (e.g., development costs, risks, prerequisites, and application scope). Places where additional research is needed are identified. While section 8.1 focuses on non-adaptive guidance approaches based on discussion scripts and representational tools, section 8.2 addresses the issue of supporting argumentation-based learning activities through automated analysis and adaptation techniques.

8.1 Representational and Script-based Guidance

The design of argumentation learning user interfaces has a major impact on whether argumentation processes and learning are effectively supported or not. The different approaches to argumentation user interfaces can be broadly classified as discussion scripting approaches (i.e., structures imposed on the communication between students) and knowledge representation approaches (i.e., structures imposed on argument representations students create). The learning arrangement developed as part of this dissertation (see Chapter 4) combines both of these approaches in a novel way.

This section synthesizes the current state of the art with respect to non-adaptive user interfaces; the main points of this discussion are presented in Table 40, p. 229. First, insights regarding discussion scripts (as a form of social scaffolding) and representational tools (as a form of representational scaffolding) are discussed separately (subsection 8.1.1 and subsection 8.1.2, respectively). Then, the issue of combining representational and social scaffolds is addressed (subsection 8.1.3). More advanced solutions for supporting argumentation learning involving adaptive and intelligent technologies, which may be used to enhance the non-adaptive user interfaces addressed in this section, will be discussed in section 8.2.

8.1.1 Discussion scripting

A discussion script is a pedagogical approach to support students in leading discussions in productive ways. There are different approaches to supporting discussions, for instance, procedural instructions, role assignments, prompts, and structured communication interfaces, which sentence openers is one possible implementation of. These different methods may be combined, for instance, by assigning roles and supporting these roles with corresponding sentence openers, which is the approach investigated in this dissertation (see Chapter 4).

This section focuses on structured discussion interfaces. The sentence opener approach, which was used in this dissertation, will serve as a leading example. Where appropriate, related approaches will be considered as well, e.g., the form-like discussion interface used by Stegmann et al. (2007, 2012) and Weinberger et al. (2010). Generally, discussion scripts are instances of collaboration scripts. Hence, the results discussed in this section may also be seen from the more general perspective of collaboration script research.

Subsection 8.1.1.1 revisits the script theory of guidance (Fischer et al. 2013) as a possible theoretical framework to account for effects of discussion scripting. It applies the script theory to the approach used in this dissertation (see Chapter 4) to exemplify aspects of the script theory in relation to discussion scripts. The following subsections review empirical results regarding the effect of structured discussion interfaces on the discussion quality (subsection 8.1.1.2), the learning of discussion skills (subsection 8.1.1.3), and the learning of subject matter content (subsection 8.1.1.4). The remaining subsections discuss miscellaneous issues, in particular, how group and process awareness may profit from script elements (subsection 8.1.1.5), the potential misuse and non-use of script elements (subsection 8.1.1.6), and adaptable discussion scripts (subsection 8.1.1.7). Subsection 8.1.1.8 summarizes the main points of this section.

Table 40

Analysis of approaches to the user interface design of argumentation learning systems

Approach	Benefits	Limitations
Discussion scripting (sentence openers; pre-structured input fields)	**educational benefits** – encourages / enforces use of high quality discussion moves – makes intention of discussion moves explicit, thus, increases group and process awareness – evidence for improved discussion quality (task-focus, critical moves, evidence and reasons, elaboration) – transfer of practiced discussion moves to unsupported situations plausible (but no direct evidence)	**educational limitations** – may be misused by students – may be ignored by students – may inhibit / conflict with students' own approaches to argumentation (internal scripts) – may cause unnecessary load if students already competent in targeted behavior – mixed evidence with respect to the learning of subject matter content
	implementation benefits – relatively low development costs for new settings once a general technological infrastructure exists – applicable to support many different scenarios (e.g., discussion types) – makes discussion move semantics accessible to automated analyses	**implementation limitations** – medium costs of iterative testing to avoid usability problems (e.g., to ensure that sentence openers are well understood, provide a sufficient variety of choices, and are in line with pedagogical objectives)
Represent. guidance (argument diagrams)	**educational benefits** – makes argument structure visible – ontologies encourage reflection on basic concepts of argumentation – helps systematically explore the space of debate – facilitates evaluation of arguments – serves as resource and stimulus in discussions – evidence for improved process characteristics (content co-elaboration, inference making) – evidence for gains in reasoning skills (arg. analysis, causal inferences, crit. think., policy deliberation, legal arg., analysis of hist. controversies)	**educational limitations** – may be hard to build and understand (if ontology is complex) – becomes unwieldy when modeling large, complex, and highly interrelated argumentative content areas – students might lose a shared focus, i.e., students work independently, build their own "argument islands" – requires considerable student efforts to maintain readability (e.g., systematic organization of contents) and modify existing structures once a diagram reaches a considerable size – mixed evidence with respect to the learning of subject matter content
	implementation benefits – relatively low development costs for specific settings once a general technological infrastructure exists – applicable to support many different domains and scenarios – clear semantics accessible to potential automated analyses	**implementation limitations** – medium costs of iterative testing to avoid usability problems (e.g., to ensure that ontology is well understandable and in line with pedagogical objectives)

8.1.1.1 Script theory of guidance

From a theoretical perspective, one may employ the script theory of guidance (Fischer et al. 2013) to explain the positive effects discussion scripts, such as sentence openers, may have. According to the script theory, learners typically have already basic knowledge about collaboration and discussion practices. For instance, an assumption of the approach presented here is that learners are already able to perform basic discussion moves, such as presenting claims, supporting claims with arguments, backing arguments with evidence, and questioning claims and arguments of other discussants. In terms of the script theory, knowledge about these basic discussion moves is present in form of *scriptlets* in the learners' mental representations of collaboration knowledge. These mental representations may be seen as *internal collaboration scripts* (Fischer et al. 2013), as opposed to external scripts that are used to influence collaboration and discussion behavior from the outside. Yet, while the basic constituents of good discussion practice are often available to learners, they may be lacking knowledge on a higher level. For instance, despite the fact that learners know the basic moves, they may nevertheless not apply these moves when engaging in discussions. What may be missing is knowledge about which specific moves are important in a particular situation to productively engage in a particular type of discussion. The script described in this dissertation (see Chapter 4) assigns to each learner two roles each pooling behaviors essential in critical discussions: being proponent and being constructive critic. The proponent role comprises moves aimed at elaborating on ideas and arguments (e.g., providing explanations, arguments, and evidence; connecting different points made before; and developing ideas further). The critic role involves moves aimed at questioning and critically examining ideas and arguments (e.g., considering alternative positions and explanations; identifying logical inconsistencies, the absence or insufficient quality of reasons; and taking counterarguments and opposing evidence into account). Thus, in terms of the script theory, sentence openers constitute *scriptlet scaffolds* that tell learners which basic moves are important with respect to the two roles. The goal of these scriptlet scaffolds is to activate corresponding internal script components; corresponding moves can then be performed by the learner independently. An aspect not addressed in the script theory is the possibility that basic skills are there but in a rather undeveloped form. So by activating internal script components, sentence openers also create opportunities to practice and, thus, refine basic discussion skills. On another level, the script conveys to learners that the behaviors represented by these two roles are important to engage in a critical discussion. That is, the script also conveys information regarding higher-level script components, in particular, information about the play *critical discussion* (a *play scaffold*) and about the roles *proponent* and *critic* (a *role scaffolds*).

The empirical results obtained suggest that the script was successful in activating internal script components that raise the quality of discussion, as indicated by higher levels of elaboration (*external script guidance principle*). The improved discussion quality also indicates that the script did provide support at the "right" level, i.e., a level that potentially leads to the acquisition of knowledge (*optimal external scripting level principle*). According to the script theory, sustained engagement in initially unfamiliar practices—which high-quality critical discussions may have been for participants of the presented study—leads to the development of higher-level internal script components (*internal script induction principle*). Therefore, the scripted learning activity may have contributed to improved discussion skills. The presented results did not provide direct evidence for such improvements, since this aspect was not directly tested for. However, improvements of the discussion quality during the intervention and positive self-assessments regarding argumentation learning suggest that positive effects on discussion skills are at least plausible. According to the script theory, CSCL practices that are transactive in nature (i.e., involve the reasoning on the reasoning of others) are assumed to lead to improved learning of the content elaborated during the CSCL practice (*transactivity principle*). The critical discussions fostered by the script certainly fall into the category of transactive CSCL, so it is plausible that learners also profited in terms of knowledge gains of discussed contents. While the presented study did not find significant differences in terms of detailed factual knowledge, students may have profited in terms of a better understanding of the space of debate (Baker et al. 2007), that is, knowledge about the relationships between the positions, claims, arguments, values, etc. relevant to the controversy of climate change—a kind of knowledge not tested for in the study.

Two potential pitfalls when employing discussion scripts follow from the script theory of guidance. First, discussion scripts might be too restrictive and hinder students to make use of effective internal script components. For instance, if students are forced to choose from a limited set of given sentence openers they may be prevented from applying other high-quality discussion moves available in their internal repertoire. Even if the use of sentence openers is not mandatory, their mere presence in the user interface may already inhibit the activation of internal script components and thus reduce chances that certain high-quality discussion moves are produced. For instance, students may too strongly focus on complying with the given script or feel not authorized to diverge from it. Second, students may already be competent in behaviors targeted by the external script. That is, they may use appropriate critic and proponent moves even without sentence openers being available. In this case, sentence openers may impair students' performance by causing distraction, unnecessary (e.g. cognitive) load, and demotivation, e.g., when students feel patronized.

The favorable outcome of the presented study show that the two potential pitfalls—interference with effective internal script and extraneous cognitive load—apparently played only a minor role, if any, in the presented arrangement. However, the presented intervention was applied for a relatively short amount of time. The two potential pitfalls may gain relevance for more long-lasting interventions. In general, if an external script is effective, students will, at some point, not be fully depending on this script anymore. They will internalize script components and develop own effective approaches to discussion. To prevent the two discussed pitfalls, it might be helpful to gradually reduce the level of support over time (i.e., fading the scaffold; Pea 2004). How this can be realized with sentence openers is an open question. For instance, suddenly taking sentence openers away during a discussion would be confusing for students and may be misinterpreted as a prompt to not use this kind of discussion move anymore. Therefore, such kinds adaptation may be most appropriate between sessions rather than within sessions. Alternatively, students may be made aware of the reasons for the presence and the fading of structuring elements to avoid confusion and wrong interpretations.

8.1.1.2 Effects on the discussion quality

Probably the main goal of sentence openers is to indicate to students which kinds of discussion moves are particularly appropriate. Thereby, sentence openers encourage students to also use these high-quality discussion moves at a higher rate. This assertion is strongly supported through the body of empirical evidence, which shows improved discussion quality in terms of task focus, critical moves, evidence, reasons, and elaboration depth and breadth (Baker and Lund 1997; Nussbaum et al. 2002; McAlister et al. 2004; Oh and Jonassen 2007). Also the results reported in this dissertation (see Chapter 4) contribute to this body of evidence, showing that students make more extensive use of elaboration moves when supported through sentence openers (*elaboration moves* used here as an umbrella term for all the characteristics of quality discussions mentioned above). In a similar vein, it has been shown that form-like discussion interfaces that explicitly prompt students to specify claim, ground, and qualification for each posted message have positive effects on the quality of arguments (Stegmann et al. 2007, 2012; Weinberger et al. 2010). All in all, these results demonstrate that structured discussion interfaces, utilizing sentence openers or other forms of structuring, can be effective in achieving one of their main objectives—to improve the quality of the discussion process.

8.1.1.3 Effects on the learning of discussion skills

The ultimate goal of discussion scripts, such as sentence openers, is that not only students' performance *during* the intervention is improved, but also that improved behaviors are internalized, maintained, and transferred to situations in which no external script is available. Here, the effect of discussion scripts is still unclear due to a lack of studies that explicitly tested for transfer effects. A few studies, however focused on other scripting approaches, tested whether training effects transfer into unsupported practice. For instance, Rummel and Spada (2005) showed that positive collaboration behaviors, supported through a collaboration macro script, transfer into an unsupported application phase. Dyke et al. (2012) tested whether students who were supported during an intervention through adaptive prompts showed improved behaviors in a subsequent classroom discussion. Interestingly, while Dyke et al. (2012) did not find significant process improvements for supported students during the intervention, they identified positive effects on learning gains and on participation in the unsupported classroom discussion. Hence, there may also be similar transfer effects for sentence openers, which have been shown to considerably influence discussion quality already during the intervention. Moreover, while not testing for transfer in terms of discussion quality, some studies found that students benefited from discussion scripts in terms of knowledge *about* formal qualities of good argument and argumentation sequences (Stegmann et al. 2007, 2012; Weinberger et al. 2010). In a similar vein, students' self-reports in the study presented here indicated that scripted students assessed their learning success with respect to argumentation knowledge and skills more positively compared to unscripted students.

In general, one should be careful in terms of what can be realistically expected in terms of measurable transfer. The intervention time in empirical studies is typically rather limited, maybe one, maybe a few experimental sessions. So the question is how much influence such a short-term intervention can have on complex higher-order skills, such as argumentation, which typically need years to develop (Osborne et al. 2004; Van Gelder 2005). Although one cannot expect that learners become high proficient arguers (i.e., ones producing arguments with high persuasive power or exhibiting sophisticated strategic argumentation skills), it should at least be possible to identify quantitative changes in the supported behaviors if the script is successful (e.g., more grounded and less bare claims without considering the quality of individual grounds). Such behavioral changes should not require extensive experience and deep understanding of argumentative practice but rather basic knowledge regarding which kinds of moves are important and appreciated. According to the script theory of guidance, learners may already be able to produce these basic moves themselves in more or less elaborated form and only need support in activating these existing behavioral components. Moreover, once learners recognize the importance of using these basic components to engage in fruitful discussions and permanently change their discussion behavior, new learning opportunities arise to bring these basic components to greater maturity, enabling learner, e.g., to produce arguments of higher logical and rhetorical quality.

In summary, the issue of transfer to unsupported situations remains an open issue, and likewise the issue of more general transfer with respect to other discussion situations, domains, or topics. It would be important to explicitly test for transfer effects of discussion scripting, and also to investigate the issue on a more longitudinal basis.

8.1.1.4 Effects on subject matter learning

Another issue that certainly needs more investigation is whether domain knowledge learning can be effectively supported. A number of studies did not show significant advantages for structured discussion interfaces in terms of domain knowledge acquisition (Oh and Jonassen 2007; Stegmann et al. 2007, 2012; dissertation results reported in Chapter 4). In contrast, other research shows that structured discussion interfaces can have such positive effects (Weinberger et. 2005; Weinberger et al. 2010). Stegmann et al. (2012) found some indirect evidence for improved acquisition of domain knowledge through discussion script. Scripted students engaged more in cognitive elaboration processes during the intervention (measured through a think-aloud protocol) and cognitive elaboration was, in turn, positively correlated with domain knowledge acquisition (measured in a post-test).

Generally, the kind of knowledge one wants to promote and expects to improve through high-quality discussions—deep and conceptual understanding in a knowledge domain (Asterhan and Schwarz 2009)—is generally difficult to measure in a standardized way. In a good discussion, students may come up with novel and unique ideas that can hardly be anticipated in a standardized test. Also, different student groups may explore different aspects of a given space of debate if the topic is formulated sufficiently broad. So the absence of clear results in this respect may be due to problems in measuring effects rather than in achieving effects through discussion scripts. Addressing the problem is up to future research, e.g., by developing and validating content analysis approaches for open, essay-style answer formats.

An alternative or complementary explanation for the absence of direct positive effects is that external scripts put addition load on the students (Fischer et al. 2013). While trying to comply with a script, students may have less mental resources available to process subject matter content. The effective learning time may be reduced and less content be covered since some time must be spend on getting to grips with the script. Indeed, some research shows that, while scripting has positive effects on argumentative knowledge elaboration, it may reduce the number of task aspects covered and impair task performance[12] (Weinberger et al. 2010). So, while improving collaboration and argumentation skills, there is the danger that students fall short on improving their content knowledge. Positive in this respect is that, in the study reported here (see Chapter 4), scripted students did at least not perform worse than unscripted students. Hence, while potentially supporting the learning of domain-general skills, the script did perceivably not impair the acquisition of domain-specific knowledge. The results of Weinberger et al. (2010) show that process advantages induced by a script (improved argumentative knowledge elaboration) can even over-compensate possible process losses (reduced task coverage and task performance), as indicated by improved learning gains compared to unscripted students. Moreover, even if immediate advantages in terms of content knowledge are not achieved, it can be expected that the acquisition of content knowledge will be accelerated in future discussions when students can capitalize on their improved collaboration and argumentation skills.

8.1.1.5 Group and process awareness

Sentence openers may not only clarify the expectations regarding how to conduct productive discussions and provide guidance for the process. Rather, they can also be employed to make the intentions of past discussion moves explicit in the discussion trace, thus, helping students to create and maintain awareness about the discussion process (Bodemer and Dehler 2011; Weinberger 2011). In the study reported in this dissertation (see Chapter 4), colors have been used to highlight proponent and

[12] In the Weinberger et al. (2010) study, *task performance* and *argumentative knowledge elaboration* were treated as two separate constructs. *Task performance* refers to the adequate application of knowledge to solve a given problem case. *Argumentative knowledge elaboration* refers to the frequency of warranted and qualified claims.

constructive critic moves in the chat, so students can get an immediate idea of what other discussants want to express. Moreover, they can see, at a glance, to what extent each discussant recently used specific move types (e.g., critical moves). Overall, awareness of individual discussion styles may improve, since one can see, and roughly approximate, the type of moves each participant prefers. Increased awareness may provide "tacit guidance" to students (Bodemer and Dehler 2011), leading them to self- and co-regulate their behaviors in order to improve the quality of the discussion. The high intra-class correlation reported in this dissertation (see Chapter 4) may be explained based on an increased group awareness induced by sentence opener highlighting. Students may have felt more obliged or motivated to use sentence openers when seeing that their partner did so as well. Seemingly, awareness of the partner's sentence opener use rate influenced their own use rate. However, this mutual attunement cuts in both directions. While some dyads converged towards a high-level use rate, other dyads stagnated at a low level of use.

An important future research question is whether and how systems should also provide explicit guidance (e.g., by providing direct, e.g., textual, feedback regarding a desired mode of discussion), rather than simply mirroring awareness information (Soller et al. 2005). Yet, while more explicit forms of guidance have a lower demand on students' self-regulation competencies, they also give students less opportunity to practice self-regulation. Moreover, explicit guidance runs danger to be perceived as annoying and interruptive by students. In this respect, providing tacit guidance through awareness support (e.g., highlighting based on the use of sentence openers) may constitute a healthy “middle ground” (Buder and Bodemer 2008) between no guidance and explicit guidance: Learners may keep aware of important features of their collaboration, including possible problems. Yet, the provided information is delivered in an unobtrusive way, without interfering with the process or depriving students from regulating their learning themselves.

As a third option, awareness information may be provided in more explicit, yet still unobtrusive form. Soller et al. (2005) make a distinction between *mirroring systems*, and *metacognitive tools*. Mirroring systems display basic information about the collaboration process as is; the highlighting of sentence openers in the approach presented may be characterized as a mirroring approach. Metacognitive tools further enrich this information about the actual state with information about a target, or desired, state. For instance, a visual meter may represent the ratio between critic and proponent moves on a scale between *consensus-orientation* and *conflict-orientation*. Ratios in the middle range may be color-coded as desirable (green), while ratios at the extremes of the scale may be color-coded as *unhealthy* (red) [for a similar approach, see Soller et al. (2005), figure 3, in reference to Jermann (2004)].

8.1.1.6 Use and misuse of script elements

A possible disadvantage of sentence openers is that students might misuse them and select sentence openers that do not adequately represent the intentions of messages (Soller 2001). In particular, when students are forced to select sentence openers while composing new contributions and none of the available options is appropriate for their needs, they may feel tempted to just select some sentence openers, no matter whether it fits the sentence or not. Conversely, when students can freely choose whether or not to use a sentence opener, they may disregard sentence openers with the consequence that the discussion script cannot take effect (Lazonder et al. 2003). That is, the potential benefits described above cannot be fully realized.

Yet, even if students do not use sentence openers, they may nevertheless benefit from the presence of sentence openers in the user interface. Sentence openers can be seen as permanently present reminders of what counts as legitimate and good discussion moves. For instance, in the FACT-2 (see Chapter 4), sentence opener buttons displayed typical proponent and critic moves in the chat interface, reminding students of their roles and associated behaviors. To ensure that students use, or at least orient themselves towards the given script elements, it is important to carefully design and test sentence opener interfaces.

In the study presented here, there was virtually no misuse of sentence openers and, overall, a reasonable rate of use, indicating that the design of the sentence opener interface was appropriate for its purposes. Another result was that student dyads largely differed in terms of use rates. While some dyads made heavy use of sentence openers, others did not make much or any use of them. Such group-dependent differences show that the used one-size-fits-all approach may be appropriate for most but not all dyads. So the application of adaptation techniques, discussed below, may be considered to provide additional support to dyads that do not follow the script.

8.1.1.7 Adaptable discussion scripts

An alternative to adaptive support in using a script may be to give students the option to adjust the level of structuring themselves (i.e., *adaptable scripts*; cf. Fischer et al. 2013). For instance, before starting the discussion on the actual topic, students may negotiate the rules of their discussion. As part of this negotiation, students may discuss which discussion moves (and corresponding sentence openers) can be considered most productive to resolve a given issue. Through such kind of discussion, students may gain important insights *about* argumentation, for instance, that it is essential to provide reasons for one's claims, that evidence is important to substantiate arguments, or that critical questions can help identify reasoning bugs and may lead to a deeper understanding of subject matter content.

A similar approach has been used, for instance, by Wegerif et al. (1999). An important component of their intervention program was classroom sessions in which students discussed and agreed on ground rules for fruitful argumentation. The resultant ground rules were then put in big letters on the classroom wall to be permanently accessible to students. In computer-mediated discussions, sentence openers may take a similar role by materializing the rules students agreed on. That is, based on their agreement, students may configure the sentence opener interface themselves by choosing the move types they deem most important and the specific formulations they prefer. Besides the pedagogical value that such activities may have in themselves, students may become more inclined to actually use sentence openers. First, they may have gained a better understanding of the purpose and application of specific sentence openers. Second, they may be more comfortable using sentence openers that correspond with their personal preferences. Third, they may feel committed to follow the rules they explicitly agreed on with their partner. An interesting question is then whether the negotiation about discussion rules and sentence opener itself can be supported and guided in some way, e.g., to ensure that representatives of the most critical categories, such as *arguing* and *counter-arguing*, are ultimately included.

8.1.1.8 Summary

The results clearly show that discussion scripts can positively influence the quality of discussion processes (Baker and Lund 1997; McAlister et al. 2004; Nussbaum et al. 2002; Oh and Jonassen 2007; Stegmann et al., 2007, 2012; Weinberger et al. 2010). The recently proposed script theory of guidance (Fischer et al. 2013) provides a powerful tool to explain and predict positive and negative effects of specific collaboration scripts, including discussion scripts. In terms of argumentation knowledge and skills, it has been shown that students benefit from discussion scripts. For instance, students performed better in post-test tasks in which they had to name the typical components of an argument or construct an argument as complete as possible (Stegmann et al. 2007, 2012; Weinberger et al. 2010). Yet, while this indirect evidence makes it plausible that students will also perform better in future discussions, a direct empirical proof of such transfer effects is still pending. With respect to domain knowledge learning, there was no discernible effect in a number of studies despite the fact that the quality of collaboration was improved (Stegmann et al. 2007, 2012; dissertation results reported in Chapter 4). Other studies (Weinberger et al. 2005; Weinberger et al. 2010), however, found positive effects in terms of the acquisition of domain-knowledge. More empirical work is needed to clarify under which conditions domain knowledge gains can be realized.

8.1.2 Representational tools

Representational tools support individuals and groups of students in creating an external representation of argumentative structures. In the simplest case, a representational tool may be a standard word processor, which allows students to compose argumentative texts. During the past decades, educational researchers turned their attention to more structured and specialized knowledge representation approaches such as argument diagramming, which is the approach used in this dissertation (see Chapter 4) and the main focus of this section.

Subsection 8.1.2.1 revisits the theory of representational guidance (Suthers 2003), an important theoretical account explaining the effects representational tools can have on reasoning and collaboration. The following subsections focus on educationally relevant aspects of the use of argument diagrams as representational tools, in particular, the effects on on-task performance (subsection 8.1.2.2), on the learning of reasoning skills (subsection 8.1.2.3), and on the learning of subject matter content (subsection 8.1.2.4). Subsection 8.1.2.5 discusses risks and limitations to consider when employing argument diagrams. Subsection 8.1.2.6 discusses argument diagrams in relation to other representational formats. Subsection 8.1.2.7 summarizes the main points of this section.

8.1.2.1 Theory of representational guidance

Argument-diagramming interfaces provide graphical languages to represent arguments in semi-structured formats. Their theoretical underpinning can be described in terms of Suthers’ (2003) theory of representational guidance. On the one hand, the graphical language imposes representational constraints. It requires from students to decompose arguments into a set of discrete knowledge chunks and relations, and to classify these knowledge chunks and relations in terms of predefined categories. On the other hand, the graphical language makes certain aspects of the representation more salient. For instance, unlike in a text, individual knowledge chunks and their relations are immediately visible in a diagram. Also, the types of knowledge chunks and relations are explicitly represented in argument diagrams. The combination of constraints and saliences leads to specific perceived affordances of the user interface (Norman 1988), meaning that the user interface suggests and triggers specific mental, physical, and social activities. Suthers and Hundhausen (2003) report empirical evidence in support of this assertion, focusing on structural properties of representational formats. Students who used highly structured representational formats (i.e., graphs and tables) elaborated more on knowledge items compared to ones who used a less structured format (i.e., text). Schwarz and Glassner (2007) provide support for the effect of ontologies on students' behavior and performance. Students who created diagrams based on a given ontology of argumentation outperformed others who were not provided with a specific ontology (i.e., unlabeled boxes and links). For instance, the ontology encouraged students to represent a larger number of claims and arguments. In fact, the Schwarz and Glassner (2007) approach can be seen as a hybrid approach between argument diagramming and discussion scripting, since students used diagrams as the discussion medium rather than as a supplement to a different discussion medium such as a chat. Therefore, the provided ontology can be seen as a kind of discussion script.

8.1.2.2 Effects on on-task reasoning and collaboration

Theoretical grounds and empirical evidence support the assertion that diagrams facilitate reasoning, inference, and collaboration during task execution. One of the main properties of representational tools like diagrams is to "make thinking visible" (Bell 1997). For instance, diagrams can visualize the amount and quality of support and opposition of claims, thus, helping students to evaluate argument components more easily (Twardy 2004). Empirical results show that students made more valid inferences in policy deliberation problems when a diagrammatic representation was available (Easterday et al. 2010). In collaborative learning arrangements, diagrams can serve as resources, stimuli, and guides for student discussions (Suthers 2003). For instance, in the study conducted as part of this dissertation (see Chapter 4), students frequently referenced diagram elements (on average, in about 12% of all messages) and followed the lines of argumentation represented in the diagram during their discussions. Diagrams can help students to systematically explore a space of debate by providing a persistent, well-organized group memory (Buckingham Shum et al. 1997).

Another insightful interpretation can be gained from a perspective of situated cognition (Brown et al. 1989). Situated cognition scholars typically analyze performance and learning in terms of activity systems comprised of the learner and the social and material surrounding. The social and material environment offers possibilities to *off-loading* parts of a task, freeing personal resources that can be invested on other, more important or more difficult task aspects. In the specific arrangement used in the present study (see Chapter 4), learners can first focus on the analysis of the argument in their text and off-load the results to the diagrams. When entering the discussion, parts of the basic reasoning have already been done, are materialized in the diagram, and therefore do not further block mental resources. Consequently, learners can put increased attention to more advanced aspects of reasoning, e.g., identifying connections between their text and the unfamiliar text prepared by their partner, and to social interactions, e.g., responding to the partner in reasonable ways.

8.1.2.3 Effects on the learning of reasoning skills

There are a number of studies that indicate that individual argument diagramming is an effective pedagogical approach in teaching higher-order reasoning skills, such as argumentation, critical thinking, and causal inference. For instance, Pinkwart et al. (2009) report on a study, in which particularly low-aptitude law students profited from the LARGO argument diagramming system in terms of improved legal argumentation skills. Other researchers used argument diagramming as integral part of University-level philosophy classes and evaluated the success over an entire semester. Twardy (2004) reports that argument diagramming significantly improved critical thinking skills compared to a control condition taught with traditional methods. Similarly, in two studies by Harrell (2008), argument diagramming significantly improved the acquisition of argument analysis skills. As discussed before, the results of Easterday et al. (2010) show that ready-made diagrams helped students to make better causal inferences. These students outperformed others who had only a text version available and ones who created diagrams themselves with a computer tool. Yet, when it came to a near transfer task, in which students analyzed new problem instances without diagrams or a diagramming tool available, those who actively created diagrams during the treatment phase outperformed the other two conditions. So while diagrams can organize information in a well-understandable way and thus facilitate reasoning tasks, it is apparently the process of actively creating diagrams that is most effective in acquiring higher-level reasoning skills for application in unsupported situations.

8.1.2.4 Effects on subject matter learning

The situation is less clear for the learning of specific subject matter content that is mapped out in diagrams. Dwyer et al. (2013) provide evidence for the effectiveness of diagrams and argument diagramming. They present a series of three studies that show that (a) students recall more information from an argument diagram compared to a text, and (b) students who construct argument diagrams recall more information compared to those who write text summaries, which is a standard technique for text reading and comprehension. Janssen et al. (2010) show that diagram users outperform those who use a list-based format in terms of test scores computed from factual and insight multiple-choice questions. However, Suthers and Hundhausen (2003) found in a text analysis task no advantage of diagrams over a note-taking tool in terms of domain knowledge learning, despite the fact that more knowledge elaboration took place in the diagram condition during the intervention. This result mirrors the observation made in this dissertation (see Chapter 4) and by others (e.g., Stegmann et al. 2007, 2012) with respect to discussion scripts, namely that improved characteristics of the collaboration process do not necessarily translate into improved domain knowledge learning. Finally, van Drie et al. (2005) found that a control condition without a representational tool significantly outperformed diagram users on several dimensions in a posttest. Yet, this result is less surprising, given that the quality of collaboration was also better for the control condition.

8.1.2.5 Risks and limitations

There are a number of potential limitations of argument diagrams to consider as well. To construct diagrams in an appropriate manner, students must learn the meaning and use of the elements of a given ontology. While this gives students an opportunity to learn about important concepts of argumentation, they may be overwhelmed with the complexity of the given ontology, leading to an inappropriate use. For instance, the initial version of Belvedere was too complex and therefore simplified in consecutive versions (Suthers et al. 2001). In the study presented here, the used ontology was deliberately kept simple and self-explaining, so only little training was required to create diagrams.

To provide an added value, diagrams must be laid out in a well-organized way, preferably according to Gestalt principles such as symmetry, continuation, and proximity (Dansereau 2005). Otherwise the diagrams may become hard to read. Therefore, students must possess the competency to organize information intelligibly using a graphical format. Moreover, ensuring an appropriate diagram layout requires time and effort that students cannot spend for the actual learning task.

Diagrams also have the problem of a "premature commitment to structure" (Buckingham Shum et al. 1997). That is, at the beginning, students typically do not have a clear idea of how the final diagram will look like. Therefore, they may make decisions regarding the structure of the diagram that are expensive to revise in later stages. For instance, a student may recognize only late in the process that an element at the left end of the diagram is related to another element at the right end of the diagram. The student could create a link that crosses the whole diagram, which would impair the readability of the diagram. Or the student could move the two elements closer to one another, which may introduce layout problems for elements linked to the two elements under consideration. In the worst case, there is no one configuration of the diagram that lays out all elements and relations nicely.

In general, the complexity of diagrams quickly reaches a point at which the diagram is hard to read and maintain in an orderly state (Loui et al. 1997; van Drie et al. 2005). So node-and-link diagrams have a limited lifetime and are only useful to represent a moderate number of elements.

8.1.2.6 Diagrams in comparison to other representational notations

Some studies indicate that diagrams have favorable representational properties over alternative formats. For instance, Janssen et al. (2010) found that a diagramming tool led to higher-quality representations and better post-interventional essays compared to a list-based format in the domain of historical reasoning. However, they did not find a difference in terms of collaboration quality. Suthers and Hundhausen (2003) found that, in contrast to graphical formats, tabular formats potentially cause students to consider weak or spurious relations when systematically checking each table cell.

Other studies again indicate that other representational formats may be superior to graphs for certain tasks. For instance, van Drie et al. (2005) found that student groups who used a tabular format engaged significantly more in historical reasoning and performed more elaboration and knowledge co-construction moves during their discussions. Even students in the control condition, who did not create any external knowledge representation, outperformed diagrams users. Yet, as noted by the authors, the positive results of the control condition may be an artifact of the used analysis approach. Students who did not create an external knowledge representation simply had more time for their discussions, and consequently, could produce a higher number of quality discussion moves. Since in online communication, diagramming activities may be carried out in lieu of corresponding discussion moves (cf. Suthers et al. 2003), an analysis should consider both, discussion moves and diagram activities.

The fact that some research pictures diagrams as beneficial for learning while other research highlights limitations of the diagramming approach may be explained by differences in the structure of knowledge students engaged with. While in some cases, knowledge has a complex, networked structure, in other cases, the inherent structure of knowledge is of a more limited nature, with only few crosslinks between individual knowledge items. So, in some cases, diagrams may provide a value added by capturing and clarifying the complex structure of knowledge, in other cases, creating diagrams may produce more overhead than added value. Empirical research is needed to clarify the connection between properties of representational notations and the complexity of knowledge structures to be represented.

8.1.2.7 Summary

It is well documented that representational tools can have a major influence on the quality of reasoning and collaboration processes and that the specific representational format makes a difference. The framework of representational guidance (Suthers 2003), with its analysis of representational formats in terms of saliences and constraints, provides a powerful tool to generate predictions and to inform the design of learning arrangements. Yet, the empirical results with respect to different representational formats are somewhat inconclusive (e.g., support for graphs in Janssen et al. [2010], support for tables in van Drie et al. [2005]). Therefore, more research is needed to decide which formats are most appropriate to support specific tasks and processes. With respect to higher-order reasoning skills, evidence consistently shows that students benefit from argument-diagramming activities (Easterday et al. 2010; Harrell 2008; Pinkwart et al. 2009; Twardy 2004). The results of Twardy (2004) give indication of transfer learning: Rather than improving reasoning skills in one specific training domain, students taught with the argument mapping method performed significantly better in a general critical thinking test compared to ones taught with traditional methods. Although the Twardy (2004) results are impressive, it must be said that the level of experimental control was relatively low, calling for more rigorous empirical investigation. Overall, the positive results with respect to reasoning skills were obtained in studies focused on individual argument diagramming. So the question is in how far the results can be extended to collaborative settings. With respect to the learning of the specific knowledge represented in the diagrams, the results are mixed ranging from positive effects

(Dwyer et al. 2013; Janssen et al. 2010) to no effects (Suthers and Hundhausen 2003) to negative effects (van Drie et al. 2005).

8.1.3 Comparison of both approaches

Discussion scripting and argument diagramming have a number of characteristics in common. On the one hand, they restrict the space of possible user actions compared to more generic tools for communication (e.g., computer chat, forums) and knowledge representation (e.g., word processors, note-taking tools). On the other hand, they make certain information more salient in the user interface compared to their generic counterparts, in particular, high-quality discussion moves and argument structures and components. They create specific affordances, thus, encouraging certain behaviors while suppressing other behaviors. The task of the designer is to build user interfaces with the "right" affordances to achieve desired pedagogical objectives. This is a very challenging task as evidenced by non-existing and negative effects observed in a number of research studies. Clearly, more empirical work is needed to disentangle the complex interplay between user interface characteristics and other relevant independent variables such as specifics of different knowledge domains, student populations, and task characteristics. From a practical point of view, the specific configurations used in discussion scripts and knowledge representation tools should be iteratively tested and refined before they are used in practice. This process can be supported through generalized frameworks like LASAD (see Chapter 5), which allow setting up user interface configurations with only little development effort and technical know-how required.

A strength both approaches have in common is that they are relatively broadly applicable. For instance, the sentence openers used in the study reported here (see Chapter 4) are not specific to the topic of climate ethics and can essentially be used to discuss arbitrary other topics. Depending on the specific pedagogical objectives and the kind of moves one wants to foster in a discussion, different sets of sentence openers may be conceived and implemented to realize specific *dialogue games* (Ravenscroft 2007) or *epistemic games* (Morrison and Collins 1996). Similarly, argument diagrams have been used to graphically represent knowledge in domains as diverse as philosophy, the Law, policy deliberation, planning and design, and science to enable or support a wide spectrum of tasks including argument analysis, argument construction, discussion, and rational decision-making. The key to such a level of flexibility is the configurability of sentence openers and argument ontologies, that is, the basic elements available to create discussion moves and diagrams.

Another strength of both approaches is that they explicate otherwise hard to capture intentions and semantics—each sentence opener typically corresponds to a specific intention and, similarly, each diagrams element corresponds to a specific semantic category. These semantics and intentions thus become easily accessible and usable by a computer program and can be automatically analyzed, e.g., to drive automated adaptations. If diagrams and sentence openers are used in combination, new interesting opportunities arise to analyze social and content-related aspects together—a topic addressed in greater depth in Chapter 9.

One important difference between both approaches is that they focus on different aspects of argumentation learning. While discussion scripts operate at the social level by shaping how students interact with one another during a discussion, argument diagrams provide a scaffold at the content level by providing a tool to organize and classify relevant domain content in an intelligible way. Being a good discussant requires both, a proper understanding of a space of debate (i.e., the relevant facts, claims, and arguments in a knowledge domain) and the competency to engage in productive discussions with others. A discussion with participants who do not have any background in a given knowledge domain is similarly aimless and doomed to fail as a discussion with participants who are not able to productively contribute their individual knowledge to advance the shared understanding of the group. This insight motivated the *Multilevel Scaffolding* approach of this dissertation (see Chapter 4), which combines argument diagramming and discussion scripts to provide optimal scaffolding on both the content and the social level. The empirical results indicate that a discussion script can provide an additional advantage to a diagram-only condition. While the question whether the diagrams provide a value added to sentence openers—or whether both approaches even reinforce one another—cannot be definitely answered based on the experimental setup, the study yielded some hints that this might indeed be the case. In particular, students used the content of diagram elements as an information source in their chat discussions, as evidenced by a considerable number of direct verbal references to the diagram (12% of all messages contained a direct reference, e.g., by mentioning a specific box number). However, in two studies investigating whether statistical interaction effects between social and content-oriented scaffolds exist, Weinberger et al. (2005) could not confirm such effects.[13] A potentially critical difference to the approach presented here is that Weinberger et al. (2005) utilized content-related prompts rather than argument diagrams. In particular, they displayed prompts like "Does a success or a failure

[13] Weinberger et al. (2005) used the term "epistemic scaffold," emphasizing that their content-related scaffolds aim at supporting task-related activities.

precede this attribution?" in the input window for new discussion board messages—an approach quite different to the one presented here.[14] A potentially important factor to create synergistic effects is whether scaffolds are designed to function largely independently from one another, or whether they are more closely connected and intertwined. For instance, in the *Multilevel Scaffolding* approach (see Chapter 4), a diagram element of type *fact* may directly prompt students to use a sentence opener *According to a statistic / estimate* (i.e., diagram element and sentence openers are based on the same *epistemological concept* [Suthers 2003]). Such connections are potentially essential to achieve synergistic effects since they may help students integrate social behavioral components (oriented towards the learning partner) and content-focused behavioral components (oriented towards the learning content) into higher-quality discussion moves. This hypothesis is in line with the view of Tabak (2004), who sees "cohesion and direct interaction between the elements of a scaffolding system" as potentially essential factors to achieve synergies. It remains an important challenge for future research to systematically investigate the factors that decide when synergistic effects can be expected. Synergistic effects would provide an even stronger argument for combining different structuring approaches than additive and sub-additive effects.

8.2 Adaptation Approaches

As discussed in the previous section, discussion scripts and external knowledge representations have been successfully used to support argumentation learning processes and outcomes. However, modern computer technology, in particular, artificial intelligence, has the potential to further improve the effectiveness of argumentation learning user interfaces through automated adaptation. A prerequisite for automated adaptation is that the software is capable of automatically analyzing student behavior to drive the adaptation process. Such analyses require computational models that can interpret observable behavior in an actionable way. There are two principle approaches, both of which investigated in this dissertation, namely to engineer models based on expert knowledge (knowledge-driven approaches; dissertation approach in Chapter 5) and to induce models from existing data through data analysis techniques (data-driven approaches; dissertation approach in Chapter 6).

[14] The learning task used by Weinberger et al. (2005) was to collaboratively analyze given cases in terms of a specific theoretical framework in social psychology concerned with attributions.

This section addresses the issue of automated analysis and adaptation. Subsection 8.2.1 discusses general considerations regarding automated adaptation and reviews empirical results obtained with specific adaptation approaches. Subsection 8.2.2 elaborates on approaches to the automated analysis of argumentation learning activities.

8.2.1 Automated adaptation

Automated adaptation has the potential to compensate for one of the main shortcomings of static learning tools, namely their inability to tailor support to the specific needs of students. The four show case applications of the CASE engine, which were discussed as part of this dissertation (see section 5.7), give a flavor of different ways how adaptation technologies can be employed to support learning activities. Adaptation may target collaboration (e.g., detected collaboration problems) as well as problem solving (e.g., relevant problem-solving steps). Adaptation can support collaboration at a fine-grained level (e.g., recommending discussion moves appropriate in the current situation) as well as at a coarse-grained (e.g., general instructions regarding the current collaboration phase). Adaptation can focus on the learning of domain-specific knowledge (e.g., critical topics not yet addressed in a discussion) or domain-general skills (e.g., faulty logical reasoning). Adaptation may alter the overall structure of the user interface (e.g., the set of available sentence openers or box types; an approach more appropriate for between-session variations due to user interface consistency reasons) or may be implemented through temporarily displayed prompts, messages, or highlighting (a form of adaptation that integrates quite naturally within individual learning sessions). Another variant is to automatically compose appropriate learning groups—a kind of adaptation that already takes place before the actual learning activity starts. Possible grouping strategies are to put together students with complementary or similar knowledge (Hoppe 1995), or different opinions (Jermann and Dillenbourg 2003).

Subsection 8.2.1.1 discusses three major promises of adaptation technologies, namely to tailor support to situation-specific needs, to individual differences, and to the learning progress. Subsection 8.2.1.2 addresses the pedagogical risk of inappropriate or misguided adaptations of the learning environment, and approaches to minimize that risk. Another important issue addressed is the costs of building adaptive argumentation learning systems and possible means to lower these costs. Subsection 8.2.1.3 turns to the pros and cons of providing content-specific support, considering both pedagogical and technical aspects. The following subsections address specific adaptation approaches used in different classes of systems and the empirical results obtained with these approaches, in particular, argument diagramming systems (subsection 8.2.1.4), educational dialogue systems focused on individual learners (subsection 8.2.1.5), and systems supporting educational discussions between two or more students (subsection 8.2.1.6). Main points of this section are summarized in Table 41.

Table 41
Analysis of approaches to automated adaptation in argumentation learning systems

Approach	Benefits	Limitations
Automated adaptation	**educational benefits** – tailoring aspects of the learning environment to individual differences, learning progress (i.e., fading of the scaffold), and situation-specific demands – improved learning experience (clever system able to understand and support) – promising first results with respect to adaptive support for argument diagramming and discussions (improvement with respect to the process and the learning of reasoning skills)	**educational limitations** – risk (and incurred costs) of inappropriate adaptations (causing confusion and frustration, impairing learning, deteriorating the credibility and acceptance of the system) – unexpected structural changes to the learning environment may cause user interface consistency problems – limited availability of empirically substantiated design guidelines how to adapt (e.g., when does an interruption cause more benefit than harm)
	implementation benefits *(depends on specific approach)*	**implementation limitations** – diagnostic capabilities of the system may not be sufficient to implement a desired adaptation approach – generally high costs of conceptualizing, implementing, and iterative testing of adaptation strategies *(also depends on specific approach)*

8.2.1.1 Benefits of automated adaptation

The most widespread form of adaptive support for argumentation learning is to provide situation-specific help. Typically, such support is delivered in response to specific problems students are not able to regulate on their own. Recognizing troublesome situations requires diagnostic competencies and an active monitoring of the problem-solving and collaboration process—skills students often do not possess or make use of (Azevedo et al. 2011). One function of adaptive support is therefore to make students aware of existing problems. Beyond that, adaptive prompts can deliver precisely the information students need to overcome problematic situations. To give an example, if the opinion of one discussant is consistently ignored, an adaptive support system could detect this situation and prompt other discussants for a reaction. A similar approach was used with the Epsilon system to trigger from students responses to unanswered questions (Goodman et al. 2005). The adaptation may not be targeted at problems but aim at invoking productive behaviors more generally. For instance, Dyke et al. (2013) incorporated the teaching strategy of revoicing into a conversational agent to support student discussions. The agent observes and analyzes student discussions, trying to associate utterance with predefined propositions. Once a key proposition has been identified, the agent posts the proposition to the chat (*So you are saying ...?*). The pedagogical rationale is to draw students' attention to key points to trigger further cognitive and social elaboration on these key points. Characteristics of the current learning situation are also in the focus of many adaptive support approaches for argumentation diagramming, for instance, Belvedere (Suthers et al. 2001) and LARGO (Pinkwart et al. 2009).

Another form of adaptive support addresses individual differences. Students differ in many respects, for instance, skill level, gender, age, cultural background, learning styles and learning preferences. Adaptation mechanisms may tailor support to such individual differences. Probably the most relevant individual characteristic is the skill level of students. Obviously, novices need more support and guidance than advanced students to engage in productive learning activities. So, one possible adaptation is to vary the amount and detail of provided help accordingly. High-performers may be ready to engage in more demanding practices. So a second possible adaptation is to lead these students to other, more advanced activities. To give a concrete example, a sentence opener interface may require from students to make a distinction between data and hypotheses. While this may be the right thing to learn for some students, others may have already mastered making this distinction. They would be ready to practice more fine-grained distinctions, e.g., the distinction between different kinds of data, such as experimental results and everyday observations. Hence, a possible adaptation may be to vary the complexity of a learning task based on the students' skill level (e.g., by increasing the complexity of choices students have to make).

Besides skill level variations between students, also the skill level of individual students is variable, expected to change over time when instructional support is effective. For instance, after using sentence openers for some time, students should have internalized how a good discussant behaves and should be able to engage in the practiced behaviors without sentence openers. Not only that the provided support may be without effect then. Rather, the gradual removal of support itself may be important for the learning process. Otherwise, students may keep relying on the external support without developing the cognitive structures needed to act as independent problem-solvers or discussants. Static support approaches in CSCL have been criticized for the absence of exactly such strategies to fading the scaffold (Pea 2004). Wecker and Fischer (2011) present evidence that fading the scaffold can improve learning outcomes significantly. Their relatively simple approach of diminishing support based on the number of student actions had a significant effect on the acquisition of knowledge with respect to the generation of counterarguments. More sophisticated adaptation approaches, driven by a model of students' actual competency, can be expected to be more effective. While skill level assessment and corresponding adaptations are at the heart of research on traditional intelligent tutoring systems, corresponding approaches for argumentation learning systems are still rare and at an early stage. Long-term student models that comprise collaboration and discussion skills are proposed, for instance, in Goodman et al. (2005) and Israel and Aiken (2007). What is up to date missing are rigorous evaluations of the accuracy and pedagogical effectiveness of such models. The main obstacle in building such models is that argument diagramming and discussion activities are more diverse and open in nature than, for instance, solving math problems, making it hard to reliably estimate students’ skill levels. The first problematic step is already the assessment of the correctness and appropriateness of individual student actions. This issue will be addressed below in greater detail.

One may hold that the provided scaffolding aims at improving the quality of discussions without pursuing the ultimate goal to teach discussion skills. That is, the scaffold is conceptualized, in the first place, as a productivity enhancer rather than as a learning support aid (Pea 2004). More productive discussions may in turn improve domain knowledge acquisition, which would be valuable from a pedagogical point of view. So, fading the scaffold would not necessarily be required then. Yet, the assumption that the scaffolding of discussions leads to an increased learning in the discussion domain is not supported through the empirical evidence reported so far. While scripts have been consistently shown to improve the discussion quality, the impact on domain knowledge learning is less clear, with a number of null results reported in the literature (Stegmann et al. 2007, 2012; dissertation results reported in Chapter 4). Furthermore, the learning of higher-order argumentation and discussion skills appears to be the more desirable goal since these skills can later function as enablers for acquiring domain-level knowledge in a wide range of different domains.

In summary, the potential advantage of adaptation approaches is that content, timing, and format of support is not predefined, based on a fixed anticipated learning process, but informed through a diagnosis of relevant aspects of the actual learning process, including personal characteristics, skills, and preferences, the learning progress made during the activity, and situational demands. Thus, a feedback loop between the learning process and the support process is established: Learning process and support process mutually influence one another. Adaptation may even be a self-reflective process. For instance, Murray et al. (2004b) present an intelligent tutoring approach that chooses tutorial actions based on an internal analysis of possible effects of the different tutorial options available in a situation. It is up to future research to investigate whether similar approaches can also be fruitful to support argumentation learning.

8.2.1.2 Risks and costs of automated adaptation

A critical risk is that the applied adaptations may be inappropriate. That is, the adaptation process may fail in that prompts and changes to the learning environment do not adequately address students' needs in a specific learning situation. This may be caused by errors and imprecisions in the computational analysis and modeling of student activities. Another reason may be that the adaptation strategy—rather than the automated analysis—has some bugs or deficiencies. So, even if the analysis of the learning process is correct, an adaptation strategy may trigger inappropriate actions in response. An inappropriate adaptation can lead to a number of negative consequences: The support of the learning process may be less effective as it could be. Even worse, the learning process may suffer from the provided "support." For instance, the learning process may be interrupted several times to display irrelevant information. Students may get confused when messages point to problems that do not exist. Or, in an attempt to fading scaffolds not needed anymore, a system may remove not yet internalized and still required helping structures. On the affective level, students may feel frustrated and lose motivation due to unnecessary interruptions. This, in turn, may negatively affect the learning process and outcomes. Finally, the experience of inadequate support may lower the system's overall credibility and acceptance. In consequence, students may even ignore reasonable hints and feedback generated by the system after such an experience.

A possible countermeasure against inappropriate adaptations is to confine support to actions that can be based on highly reliable information. That is, system developers may only incorporate diagnostic modules they deem highly accurate. Alternatively, system developers can try to design adaptation approaches in a more fault-tolerant way, to be at least not harmful to the learning process. For instance, the wording of text messages may use a suggestive rather than an authoritative tone to avoid confusion on the part of the student when the message content does not perfectly match the current situation. This approach has been widely used in both discussion-based systems (e.g., Dyke et al. 2013; Goodman et al. 2005) and argument diagramming systems (e.g., Pinkwart et al. 2009; Suthers et al. 2001). A possible practical reason why system developers try to cope with less (but still reasonably) precise diagnoses, rather than fine-tuning the analysis procedures to the limit, is that further improvements can come at high development costs—costs potentially disproportionate to the gains one would achieve in terms of the pedagogical effectiveness of the system. Even if the diagnostics are not perfect and sometimes fail, they may, by and large, still be good enough to effectively support students. Moreover, some degree of uncertainty may be unavoidable since discussion and argumentation are ill-defined domains (Lynch et al. 2009). Even human experts often disagree in their assessments regarding the correctness and quality of arguments. For instance, studies show that also experts sometimes struggle with the hypothesis / data distinction (Schank 1995). Similarly, the question whether a given proposition can stand on its own as a "known fact" or needs addition support and explanation depends on the specific community an arguer engages in. Thus, there is often no clear definition of what counts as correct or incorrect, in particular, when it comes to argumentation-related tasks. Consequently, the system's assessment necessarily entails some degree of uncertainty.

In the light of imperfect diagnostics, it might be better to build systems that adopt the role of a "fallible collaborator" rather than "feign[ing] omniscience" (Self 1990) in order to not compromise the system's credibility. Self (1990) proposes to generally use a style of support that diverges from the classical intelligent tutoring approach, which typically focuses on pinpointing and remediating behaviors the system recognizes as deficient or buggy. Often, ITS systems employ a closed-world assumption, that is, everything outside the system's model of correct behavior is considered incorrect. In consequence, the system considers each deviation from preconceived pathways as a bug and responds accordingly. Self's alternative suggestion is that systems should rather focus on guiding learners to elaborate on ideas and knowledge themselves. For instance, this may be achieved through methods of Socratic questioning or by assigning the system the role of a devil's advocate, who challenges the learners' beliefs and claims. In collaborative settings, the system may draw the attention to points in which learners potentially disagree with one another and leave the resolution to the learners themselves. To avoid confusion—e.g., when the assumed conflict of opinion does actually not exist—the system may overtly indicate its uncertainty (e.g., "*If I understood right*, you have a different opinion on *X*. For what reasons do you think your opinion is more correct than your partner's?"), or the system may not disclose at all that it assumes a conflict to exist and just try to draw the learners attention to the issue in question (e.g., "Have you already discussed *X*? What are your opinions on this topic?"). The approach to identify differences in student solutions to stimulate fruitful interactions has been used, for instance, in the COLER system (Constantino-González et al. 2003) and Collect-UML (Baghaei et al. 2007). As remarked by Self (1990), there are important reasons beyond technical pragmatism for such approaches. First, systems avoid taking an absolutist stance, which is underlying many ITS systems (i.e., entailing the assumption that absolute knowledge about the world exist). Absolutist epistemologies often lead to dogmatism, and thus are counter-productive to argumentation (Kuhn 1991). Second, such approaches are well in line with the principles of constructivism, which emphasize the value of active knowledge construction and co-construction on part of the learners, rather than just doing what a system has told to do. Finally, such approaches give more emphasis to students' self- and co-regulation competencies since the system provides a pointer but the problem is ultimately addressed by the students themselves.

In summary, system developers can try to avoid inappropriate adaptations by optimizing the diagnostics (i.e., decreasing the likelihood of false alarms) or can design adaptation strategies in a more fault-tolerant way (i.e., decreasing the pedagogical costs of a false alarm). Optimizing the diagnostics has some natural limitations in terms of what is possible (uncertainty inherent to some domains; technological limits such as the current state of the art in natural language processing) and what is affordable (development costs). Strategies for making adaptations more fault-tolerant, such as using suggestive, less definite messages rather than authoritative ones, may have a pedagogical value in themselves, e.g., from a constructivist perspective. In general, the development of effective adaptive approaches is a non-trivial endeavor, which involves significant costs and efforts in terms of conceptualizing, designing, implementing, and testing such functionalities. To find a good trade-off between the chances and risks of adaptation, but also to optimize the overall effectiveness of the system, system designers typically have to conduct multiple design-test-redesign cycles following an iterative design approach. Tools like the CASE engine (see Chapter 5) can support this iterative design process by offering configuration mechanisms that allow testing and comparing different adaptation approaches at relatively low costs.

8.2.1.3 Pros and cons of content-specific support

A crucial decision is whether support should consider the specific contents students are discussing or dealing with. For instance, a relatively lightweight and generic approach may be to design a system that only comments on the collaboration process, agnostic of the specific contents under discussion. On the one hand, this clearly involves less development efforts since there is no need to build a content-specific knowledge base and analysis mechanisms to identify knowledge chunks in student activities. Moreover, the system is better portable, since it does not depend on the existence of a content-specific knowledge base—essentially, a content-aware support approach would require knowledge bases for every new subject matter domain. On the other hand, content-specific knowledge may be a key for effective support. Walker et al. (2011) used adaptive prompts that combined interaction help (i.e., generic scaffolds for the collaboration process) with cognitive help (i.e., information relevant to the current problem instance). This design decision was motivated by the observation that students often ignore messages that provide only interaction help without a reference to the actual problem instance, possibly because these messages appear less relevant to the students. This is in line with research on the effectiveness of different e-moderation styles. Asterhan and Schwarz (2010) qualitatively compared the effect of generic, low-content, and content-specific moderator moves. They found that generic messages were neither appreciated much nor effective in terms of the elicited responses. Asterhan and Schwarz (2010) hypothesize that messages appear more salient, and moderators more involved and less detached, when a concrete connection to currently discussed contents is made.

A possible approach to lower the implementation costs of content-specific support is to architecturally separate content-specific and content-general parts of a system. This approach allows reusing the content-general part across different problems or even domains. For instance, the Rashi system (Woolf et al. 2003) decomposes the support functionality into content-specific knowledge bases and a generic inference mechanism, which provides reasoning capabilities applicable to arbitrary knowledge bases. Knowledge bases are defined in terms of relevant propositions and their relationships. The generic inference mechanism contrasts student solutions with a given knowledge base, identifies pedagogically relevant differences, and generates advice accordingly. Similar approaches to generate problem-specific support have been used in Belvedere (Suthers et al. 2001) and LARGO (Pinkwart et al. 2009). To make such an approach work, the different knowledge bases must use the same modeling primitives and semantics to be amenable to a generic inference mechanism. The engineering of consistent knowledge bases itself can be a challenging task.

Authoring tools can support the process of creating, maintaining, and managing content-specific knowledge bases (Murray et al. 2004). Such a content-focused authoring tool would be a possible enhancement to the CASE engine, which is already able to detect content-specific patterns, but does not yet provide explicit authoring support for this aspect. For instance, such functionality may involve support for (a) defining knowledge chunks for different problem instances, (b) defining inference rules that can be applied to all problem instances that share a specific ontology and obey the same domain rules, and (c) general managerial functionalities to organize created resources (i.e., knowledge chunks and inference rules). Such functionalities would extend the basic concept of LASAD—providing support across domains—to the aspect of problem-specific support, and thus, constitute a further step beyond the current state of the art. The topic of content classification and content-specific support is equally relevant to discussion-oriented systems. A number of systems implement approaches to identify discussion topics (e.g., Goodman et al. 2005; Kumar et al. 2007). Essentially the same problem is addressed in research on tutorial dialogue systems (e.g., Graesser et al. 2001).

8.2.1.4 Adaptive support for argument diagramming

On theoretical grounds, for the reasons discussed above, possible merits of adaptation in terms of learning support are highly plausible. The crucial question is therefore not if adaptation can be beneficial but rather how adaptation technologies can be used to achieve positive effects. Unfortunately, research on automated adaptation for argumentation learning is up to now meager, with only a few results reported in the literature.

With respect to adaptive support in argument-diagramming environments, the dominant approach, used in systems such as Belvedere (Suthers et al. 2001) and LARGO (Pinkwart et al. 2009), is to provide feedback only on request. This design does not require a computational approach to decide the timing of feedback and thus avoids one of the most difficult problems. Finding the right time for feedback is so difficult because students typically perform a series of actions on the diagram to accomplish a target goal. Since the system does not know the current goal a student is after, it is hard for the system to decide when the series of actions is concluded and the diagram ready for being checked. The simplest solution to avoid unnecessary and potentially harmful interruptions of the learning process is to leave the control over the timing of feedback to the students themselves. Yet, the results achieved with this approach in studies with Belvedere (Suthers et al. 2001) and LARGO (second study in Pinkwart et al. [2009]) are discouraging since students rarely took initiative to request feedback even if they would have benefited from it. Yet, even if the feedback provision approach of these systems is suboptimal, empirical results suggest that the delivered feedback messages were, in fact, helpful. The first LARGO study showed that particularly low aptitude students benefited from using the LARGO system. While the study does not allow strong conclusions regarding the effectiveness of adaptive support—adaptive support was not isolated as an independent experimental factor—a comparison with the second LARGO study is instructive. The first study (with positive effects) and the second study (without positive effects) differed considerably in terms of how often students invoked the advice function, with advice being requested far more frequently in the first study (the one with positive effects). One possible explanation is that the advice indeed helped students to better understand learning contents they struggled with. This assertion receives further support from a significant correlation between advice request frequency and learning gains in the second study.

In conclusion, empirical evidence suggests that adaptive advice in single-user argument diagramming systems has positive effects on learning. Future studies should isolate adaptation as an independent experimental factor to tease apart the effects of the basic environment, advice content (which may be provided statically or adaptively), and adaptive advice provision. The approach to provide advice only on demand has shown to be problematic. The best designed system advice is worthless if students do not take a chance to make use of it. Therefore, future research should investigate approaches to provide advice when it is actually needed—even if not explicitly requested by the student. Finding the right time to present the right advice is a challenging research problem, which involves both the consideration of the current solution state (i.e., whether the solution is ready to be checked) and learning process (i.e., whether interrupting the student's current activity causes more harm or benefit). Besides timing, another question is how to make students aware of available advice. In particular, one can imagine forms of presentation that are more obtrusive (e.g., displaying the advice directly in a popup window, which must be "clicked away") and others that are less obtrusive (e.g., a short notification with a link to the actual advice, displayed at the periphery of the user interface and possibly accompanied by a gentle sound). Important research questions concern the trade-off between, on the one hand, awareness and use of advice, and on the other hand, distraction from the task caused by the specific way how advice is presented. This issue will be addressed in greater detail below when discussing adaptive support for collaborative learning activities.

8.2.1.5 Adaptive support in educational dialogue systems

Another approach to support argumentation learning is building computer systems able to engage students in educational debates. Yuan et al. (2008) present such an approach based on computational dialectics and, more specifically, *dialogue games*. A dialogue game formally specifies dialogues as a turn-based, rule-governed exchange between opposing parties. Participants choose from a predefined set of eligible *dialogue moves*. *Dialogue rules* define legal ways of responding to previous moves. Each participant's *commitment stores* represent the positions he or she publically committed to during the debate (e.g., accepting or rejecting a specific statement). *Commitment rules* define the effects dialogue moves have on the participants' commitment stores. Some dialogue games also employ *winning and losing rules* to define conditions under which participants win or lose a debate. For instance, a party *A* may be considered as the winner of the debate when the opposing party *B* has conceded to a set of statement from which *A*'s original thesis follows with logical necessity.

Yuan et al. (2008) implemented an educational dialogue game in which individual learners argue against a computer-based agent. The user interface comprises a structured chat interface and a display of the current state of the learner's and the computer's commitment store. The chat tool allows learners to compose new messages from two sources. First, they choose from a list of predefined dialogue move types (in particular: assertion, question, challenge, withdrawal, resolution demand). Second, they select a piece of propositional content from a collection of predefined choices (e.g., "Capital punishment is acceptable" or "Capital punishment stops murderers from killing"). In contrast to the sentence opener approach used in this dissertation (see Chapter 4), the different move types have a very specific and formally defined meaning, which is encoded in commitment and dialogue rules. For instance, assertion moves add a proposition to the move maker's commitment store while withdrawal and challenge moves lead to the removal of the proposition from the commitment store; question moves ("Is it the case that *P*?") require the addressee to take a stance on the requested proposition ("*P*." "Not *P*." Or: "No commitment *P*."), etc. The behavior of the computational agent is realized by a number of rules that define the dialogue strategy of a "partially honest" agent. This means that, on the one hand, the agent does not insist on points that turned out wrong during the discussion but, on the other hand—for the sake of the argument—the agent does only concede to the learner's points if not avoidable. To produce contributions with meaningful content, the agent employs a knowledge base comprising a set of possible propositions and their relations. Propositions may support or contradict one another. Moreover, propositions are classified as hard evidence or (disputable) opinions. Developing autonomous computational agents able to maintain a mixed initiative conversations with a learner is certainly one of most ambitious ways to building adaptive computer-based learning systems and involves a number of practical limitations. For instance, one main limitation of the approach by Yuan et al. (2008) is the restricted nature of input facilities. To avoid the thorny problem of natural language understanding, the range of possible messages is fully predetermined by the available move types and propositions.

As can be learned from tutorial dialogue systems focused on content learning rather than argumentation, more free forms of interaction can be achieved based on natural language understanding technologies. Two prominent systems, each exemplifying one principal approach to the problem, are discussed next: BEETLE II (Dzikovska et al. 2014) and AutoTutor (D'Mello and Graesser 2012).

BEETLE II (Dzikovska et al. 2014) engages students in tutorial dialogues about concepts of basic electronics and electricity within a simulation environment for electric circuits. It employs *symbolic* natural language processing techniques to interpret and generate textual messages. The symbolic approach involves rule-based natural language parsers and reasoning components to construct detailed semantic representations of student utterances. BEETLE II first parses an utterance to create a domain-general representation, including the pragmatic function of the utterance (i.e., used speech acts), its grammatical structure, and word-sense information determined based on a lexico-semantic database (such as WordNet; Miller 1995). In a second step, BEETLE II employs mapping rules to translate domain-general representations into domain-specific ones. A domain reasoner can then operate on these domain-specific representations, e.g., to determine the correctness and completeness of student responses or to generate answers to student questions. Based on the diagnosis of a student answer (e.g., presented answer parts are correct but important parts are still missing), the system chooses an appropriate tutorial tactic (e.g., give positive feedback, acknowledge the correctly mentioned answer parts, and present a "keep going" prompt to encourage the production of still missing parts).

To evaluate the quality of language understanding within BEETLE II, its interpretations were mapped to five classes (student response is correct, incomplete, contradictory, etc.) and compared to corresponding judgments of human experts. The evaluation yielded an overall accuracy of 66%. Notable differences were found depending on the questions students responded to. Responses in which students had to explain a given phenomenon yielded clearly lower accuracy scores than those in which they just had to identify some target object or attribute (47% versus 88%, respectively).

Overall, the symbolic approach allows a very detailed level of language understanding and the flexible generation of system responses. On the downside, due to the high demand on upfront knowledge engineering effort, corresponding systems are expensive to build and are restricted to relatively narrow content domains. For instance, BEETLE II dialogues revolve around questions of a rather restricted scope, such as the conditions under which a bulb in an electric circuit will be on or off. The evaluation results highlight the complexity and difficulty involved in deep language analysis: Despite the relatively narrow scope of the discussion domain, BEETLE II struggles when it comes to interpreting more complex responses involving explanations.

AutoTutor (D'Mello and Graesser 2012) is a tutorial dialogue system used in content domains such as Newtonian physics and computer literacy. The dialogue starts with the tutor asking a main question (e.g., how a given observation can be explained based on the principles of physics), which is followed by an initial response by the student. The tutor evaluates the student response, gives short feedback accordingly, and initiates an extended dialogue sequence in which tutor and student jointly try to improve the initial answer. During this dialogue sequence, the tutor successively selects answer components not or not fully included in the student answer and tries to elicit these components from the student. The goal is that, to the extent possible, students themselves construct a correct and complete answer to the main question. This is realized through a series of tutor moves in which the level of scaffolding is gradually increased: (1) *pump* (e.g., "What else?"), (2) *hint* (e.g., "What about *X*"), (3) *prompt* (e.g., "*X* is a type of what?"), and ultimately (4) *assertion* (e.g., "*X* is of type *Y*"), if the student is not able to produce the target answer component himself despite scaffolding. A typical AutoTutor move comprises three parts: (1) short (positive, negative or neutral) feedback with respect to the student's last production, (2) scaffolding of one of the types described above, and (3) a cue indicting that the tutor's turn is over and the student has the floor again. Besides eliciting answer components from the student, AutoTutor tries to remedy detected misconceptions and to answer student questions.

In contrast to BEETLE II, AutoTutor mainly relies on *statistical* natural language processing techniques to interpret student responses, in particular, *latent semantic analysis* (LSA; Landauer et al. 1998). LSA is used to compute a similarity score between "good" answer components stored in the system's database and the cumulative student response (i.e., the concatenation of all student turns within the current dialogue). If the similarity score exceeds a predefined, empirically determined, threshold value, the answer component is assumed to be covered in the student response. Similarly, the system can identify “bad” answer components indicating misconceptions. The comparison of texts using LSA is based on a *semantic space*, which is computed based on a corpus of documents from the domain under consideration. The semantic space is determined based on the co-occurrence of words in documents of the given corpus. Basically, the meaning of a document is defined by the words it contains and, vice versa, the meaning of a word is defined by the contexts (i.e., the documents) it occurs in. For instance, the semantic space used in one version of AutoTutor was built from two textbooks on computer literacy. Since textbook sections typically focuses on particular content aspects, each section can be considered as one document. Technically, the computation of the semantic space utilizes singular value decomposition (SVD), a mathematical method to reduce the dimensionality of data. For instance, the 10,000 or more different words typically contained in a text corpus may be compressed to a few hundred semantic categories induced by SVD. As products of a mathematical procedure, these semantic categories typically do not have an obvious, human-understandable interpretation (therefore “latent” semantic analysis). Some research in context of AutoTutor suggests 200 as a reasonable number of dimensions. Once a representation of the semantic space is available, the similarity of texts can be determined by mapping corresponding word vectors into the semantic space and computing the distance between these vectors.

A critical advantage over simpler approaches based on keyword overlap is LSA’s ability to detect semantic similarities even if texts do not share a single word. Evaluation results show a moderate agreement between AutoTutor and human experts in identifying predefined answer components (correlations between $r = .35$ and $r = .50$; D’Mello and Graesser 2012). The upper-end performance is comparable to the agreement between intermediate human experts ($r = .49$) but clearly lower than the agreement achieved by accomplished experts ($r = .78$; Graesser et al. 2000). In comparison to the symbolic language processing approach of BEETLE II, AutoTutor does not need a formal semantic description of a target domain, nor does it need computational means to infer formal semantic representations from natural language input. Rather, a domain-specific background corpus and representative examples of good and bad answer components suffice to approximate the content of student answers. This makes the approach less expensive and also suitable for more complex domains for which domain reasoners are hard or impossible to implement. On the downside, the interpretation result is also less detailed and substantive. A particular weakness of LSA is the non-consideration of important aspects such as the syntactical structure, word order, and logical expressions, which may drastically alter the meaning of an utterance (e.g., negations).

While the described approaches of BEETLE II and AutoTutor showed effective for content learning, it remains an open challenge to develop (single-user) tutoring dialogue systems for argumentation learning. To provide a realistic context for practicing argumentation skills, a natural scenario is that the computer agent takes the role of a debater, similar to the approach of Yuan et al. (2008). The automated analysis of free-form natural language arguments is a complex endeavor, requiring a relatively detailed understanding of the inner structure and logic of utterances, e.g., to identify claims, grounds, and their interrelations. In terms of analytical complexity, arguments are more akin to explanations than to factual statements about objects and relations. As learned from BEETLE II, the analysis of exactly such explanations turned out to be particularly difficult for the system to accomplish. Moreover, the analysis of argumentation is even more challenging in some respects. Systems like BEETLE II and AutoTutor base their assessments on explicitly stated answer components and try to elicit components not mentioned yet. Argument components and relations, however, are often not made explicit but inferred by discussants based on the pragmatic context of the dialogue. Moreover, to make sense of a contribution, it does typically not suffice to focus on the contribution itself. For instance, counterarguments do often not repeat the claims they are responding to but only implicitly refer to previous contributions (e.g., *But* ...). Thus, an overall understanding of the response structure of the dialogue is needed including the specific types of rhetorical and argumentative relations that connect different contributions.

Another problem is the scope of the domain addressed. Systems like BEETLE II and AutoTutor are designed to support the learning of a predefined set of domain concepts. The implemented dialogue strategies to elicit specific answer components quite naturally restrict the discussions to content elements modeled in the systems' knowledge bases. The approach is similar to intelligent tutoring systems not involving dialog, which typically try getting students back on a right solution path again to avoid student inputs the system is unable to understand (Corbett et al. 1997). Hence, BEETLE II and AutoTutor allow mixed-initiate dialogues only to a limited extent: While also students can take initiative and ask questions, the specific topics discussed are always determined by the tutorial agent. For argumentative dialogues it may be desirable to allow students to also creatively invent new arguments, possibly drawing from background knowledge not modelled in the system. A tutorial dialogue system would not be able to understand and thus respond to such contributions (in an informed way).

Therefore, a possibly more appropriate approach is to set up the discussion between two or more students, whose behavior is not as restricted as the one of the computational debaters discussed above. Such a scenario may be supplemented with a computer agent who provides support and guidance to the discussion. The focus then essentially shifts from student-system interactions to student-student interactions. The demands on the computational agent are much lower, since the agent is not responsible for keeping the dialogue going in the first place, but rather acts as an outside observer who only intervenes occasionally. In particular, more limited analytical capabilities may be sufficient, both in terms of discussion scope (i.e., there might be discussion sequences which the agent does not understand in full detail) and accuracy (i.e., the impact of interpretation errors are less harmful since learning is primarily the result of student-student interactions). On the other hand, as discussed below, such scenarios involve a number of new, widely unexplored challenges.

8.2.1.6 Adaptive support for collaborative educational dialogue

Only recently, researchers began to investigate the specific demands for adaptive support in collaborative environments (Kumar and Rosé 2011; Walker et al. 2011). Since only few results are available, one may try to re-implement approaches known to be effective to support individual learning (i.e., one-to-one interaction between a student and a human / machine tutor). A considerable body of literature exists in this area (for an overview, see Shute 2008). However, such approaches do not readily translate into collaborative settings (Walker 2011) due to differences in the nature, dynamics, and complexity of the process. For instance, interventions by a computer agent may interrupt the flow of natural interaction between collaborating students. Empirical research shows that students tend to ignore system-generated feedback when engaged in collaboration with others (Kumar et al. 2007). Moreover, collaborative interactions have additional demands with respect to an automated analysis. For instance, a conversational agent who participates in a multi-party discussion must disambiguate the intended addressee of student messages (some other student? the tutorial agent?), while a single user tutorial dialogue system can safely assume that always the tutorial agent is addressed (Kumar and Rosé 2011). Another source of information may be research on human feedback strategies, e.g., in classroom settings (for an overview, see Hattie and Gan 2011). Yet, while some of these results can provide valuable input, guidance, and inspiration, the "lessons learned" do not necessarily transfer into settings in which the computer takes the role of the facilitator. Computer programs do not have the same level of authority and credibility as compared to human teachers. Their capabilities in understanding and interacting with humans are perceivably limited and error-prone. So students may not give the same consideration to system-generated feedback and advice. Indeed, Rosé and Torrey (2005) observed that students respond differently to questions depending on whether they are presented by a human or a computer tutor. In particular, answers to questions posed by a computer tutor were far less elaborate than those in response to a human. Often, responses to the computer tutor consisted of only a single word without further explanation or justification. In fact, the actual responses delivered in the human tutor condition were also computer-generated. That is, already the expectation regarding the dialogue partner—human or computer—has significant impact on the behavior and thus potentially on learning. In summary, while neighboring fields, such as research on traditional intelligent tutoring systems and human feedback, can be informative to the design of adaptive CSCL systems, there is a strong need for more original research that considers the particularities of CSCL arrangements. The paragraphs below discuss some of the empirical results available.

Baghaei et al. (2007) conducted an evaluation study with Collect-UML, a UML modeling tool that students use to collaboratively create and discuss UML diagrams. They compared a version of Collect-UML that only provides domain-level feedback with another version that provides, in addition, feedback regarding the collaboration process. Among other things, the collaboration feedback in Collect-UML encourages students to explain and justify changes to the group solution and to discuss differences between individual solutions created in advance to the collaborative phase. The conditions for providing feedback are encoded in a constraint-based model of ideal collaboration, which is checked in regular intervals to trigger corresponding feedback if constraints are violated. For instance, the system may detect that a change was made to the group solution without providing a justification in the chat. The collaboration support was successful in stimulating students to contribute significantly more individual solution elements to the shared group solution. While there was no significant difference in terms of domain knowledge gains, students who received collaboration feedback performed significantly better with respect to knowledge about effective collaboration. A study reported in Diziol et al. (2010) investigated whether adaptive support, comprised of a domain-knowledge and an interaction support component, is effective in supporting math problem solving and learning in a *peer-tutoring* scenario. In peer tutoring, one student takes the role of a tutor and the other student takes the role of a tutee; the roles may be swapped after some time or for new problem instances (reciprocal peer tutoring). The support of the computer tutor is based on an analysis of the support the peer tutor provides to the peer tutee. If the peer tutor marks a correct tutee answer as wrong, or vice versa, an incorrect answer as right, the computer tutor sends a message to assist the peer tutor in generating a correct feedback message. While the study yielded a null result with respect to domain knowledge learning, qualitative evidence suggested improvements of the collaboration process. A follow-up study yielded similar results (Walker et al. 2011). Again, the adaptive support did not lead to improved domain-knowledge learning, but the quality of peer tutor support was better. More specifically, there was a significantly higher rate of *conceptual help*, as opposed to *instrumental help* (i.e., help that enables the accomplishment of the next problem-solving step, yet, without explaining the rationale of that step).

In contrast to the studies discussed so far, a number of other studies show positive effects of adaptive support on domain-knowledge acquisition. In a setting similar to the one described above, Walker et al. (2011b) did find a significant advantage favoring the adaptive support condition in terms of domain-knowledge learning. To confirm that the adaptation itself makes the difference—rather than a change in students' expectation and attitude when they think that the feedback is adaptive—a secondary control condition was used in which students were told that actually randomly selected hints are based on automated adaptation. While students in the *real-adaptive condition* significantly outperformed the primary control condition, students in the *told-adaptive condition* did not so. Thus, the effect can be attributed to the actual adaptation rather than on changes in the students' expectation and attitude. Kumar et al. (2007) employed adaptive conversational agent technology to support student chat discussions in context of a mechanical engineering problem. Their approach was based on knowledge construction dialogues (KCDs), that is, pre-authored tutorial dialogues that guide students through directed lines of reasoning. Their computational agent listened to student conversations, identified domain-specific keywords, and engaged students in KCDs when relevant topics have been detected. In addition, the agent prompted the less active student in certain intervals to discuss not yet covered topics. The results show a positive main effect for the adaptive support conditions. Dyke et al. (2013) present a somewhat lighter approach compared to KCDs to support student conversations. Rather than trying to engage students in a multi-step interaction to elicit directed lines of reasoning, their approach tries to mimic somewhat simpler teaching strategies known to be effective from classroom-based settings. In particular, they employed the framework of

Academically Productive Talk (APT; discussed before under the label of *Accountable Talk*), which has been developed and researched over a period of more than 15 years (Resnick et al. 2010), to support discussions between 9th grade biology students. They conducted a study in which a conversational agent mimicked two teaching behaviors of the APT framework, *revoicing* (i.e., the agent restates the reasoning of students in different words) and *feedback* (i.e., the agent gives public praise to students who use APT moves [e.g., explanations, challenges] and encourages responses). Revoicing had significant positive effects on learning gains and quality of produced artifacts (explanations student groups came up with during their conversations); feedback had no significant impact on learning gains and even a significantly negative effect on the quality of the explanations produced during task execution. Generally, such a negative result should not be over-interpreted since, as discussed before, the approach itself may be appropriate but the underlying diagnostics or specific realization may be troublesome. However, in this specific instance, previous research has led to similar results, indicating that praise may harm the learning success (Shute 2008). As a possible explanation it was proposed that praise shifts the attention of learners to their self, distracting from the actual task.

Not only the task-related content of messages is important but also the way how messages are delivered. Walker et al. (2012) reanalyzed the data of the Walker et al. (2011b) study and found that the crucial factor for learning was the amount of relevant feedback students *took notice of*, rather than the overall amount of relevant feedback provided. Of course, when feedback is selected by the system in an informed way, the overall amount of relevant feedback is likely to be higher compared to randomly selected feedback. Thus, students also have more opportunities to notice relevant feedback. Therefore, system developers should design feedback that is both relevant and likely to be noticed. The later point may be a critical bottleneck and limiting factor, as observed by Kumar et al. (2007). They found that students often ignored contributions of a conversational agent and focused on their learning partner instead. This motivated a series of follow-up studies that investigated whether prompts that are socially more appealing and involving may also be pedagogically more successful. One study investigated the effect of agent behaviors exhibiting a sense of humor and interest in the students (Cui et al. 2009; Kumar et al. 2007b). The conversational agent asked light-hearted questions to introduce new arithmetic problems. The results yielded (weak) evidence in favor of the social condition: The working atmosphere was friendlier (e.g., no insults) and the learning results consistently better (yet not at a significant level). Kumar et al. (2010) compared three conditions that involved different levels of social engagement (task-level support was identical across conditions): (1) no social support, (2) computer-provided socio-emotional support, and (3) human-provided socio-emotional support. Social support consisted of showing solidarity (e.g., praise student contributions, encouraging participation), showing tension reduction (e.g., expression of feeling-better after a phase of work pressure), and showing comprehension and agreement with students' contributions. While the results revealed that students assessed the social behavior of the human most positively, also the computer-provided social support lead to significantly improved knowledge gains compared to the socially ignorant control condition. In a collaborative idea generation task, Kumar et al. (2011) found that the above described approach of socio-emotional adaptation improved students' perception of the agent (more supportive, less pushy). As a second factor, they compared slight grammatical variations of the feedback formulations: a *heteroglossic style* (i.e., the agent implicitly acknowledges the existence of alternative perspectives) versus a *monoglossic style* (i.e., the agent speaks matter-of-factly). The heteroglossic style improved the perception of the agent (the agent was better liked) and was more effective in terms of stimulating idea generation (more ideas produced). The results of Kumar et al. (2007), which indicated that agent-led knowledge construction dialogues (KCDs) improve domain learning, motivated further studies by Chaudhuri et al. (2008, 2009). Chaudhuri et al. (2008) investigated whether the contents that were delivered through KCDs can also be presented in a less interactive and therefore less interruptive way. They found that the non-interactive support conditions—in particular, pointers to the relevant part of a booklet and non-interactive mini-lessons (i.e., the agent just posted relevant booklet contents into the chat)—significantly outperformed the KCD condition. In line with the Kumar et al. (2007) results, each adaptive support condition, including KCDs, was significantly better than a control condition in which identical content was provided statically. Chaudhuri et al. (2009) experimented with a combination of pointers and KCDs. Rather than starting KCDs immediately, the agent provided a pointer hint first and then invited students to start a KCD, which they could accept or decline. The results provide evidence that the combination is more effect than either of the two approaches alone (pointers only and KCD only). The combined condition was the only condition with significant gains, and significantly better than a non-adaptive control and the KCD-only condition; the difference between the pointer-only condition and the pointer+KCD condition was not significant.

In summary, this series of studies shows that adaptive feedback in collaborative learning settings can improve the quality of collaboration as well as the acquisition of domain-knowledge. A number of different approaches have been employed, ranging from knowledge construction dialogues, a technique imported from single-user tutorial dialogue systems, to approaches more specifically tailored to the needs of collaborative settings, e.g., ones based on the Academically Productive Talk framework, which originates from longstanding research on classroom discussion facilitation. Research shows that subtle differences in how the feedback is delivered, such as slight variations in the feedback formulations, can have significant impact on its effectiveness. Moreover, the results suggest that feedback should not only be designed with task-related aspects in mind, e.g., to support problem solving and collaboration, but should also take socio-emotional aspects into consideration, e.g., to create a pleasant and thus possibly more productive working atmosphere. Some results indicate that the recently questioned interaction granularity hypothesis—"the effectiveness of tutoring systems […] increases as the granularity of the interaction of the system decreases" (VanLehn 2011)—is particularly problematic for collaborative learning, since interactions triggered by the system may interfere with natural interactions between human learners. Similar to argument diagramming, collaboration and discussion are typically continuous activities, making the decisions if, when, and how to interrupt an important research challenge.

Overall, research on adaptive support for collaboration is still in an early stage and a solid knowledge base for designing adaptive CSCL systems is just emerging. To systematically explore the design space for adaptive CSCL systems, researchers should conduct parametric evaluations that isolate the effect of specific design decision regarding adaptive support, as opposed to evaluating tools in their entirety (Walker et al. 2011). From a methodological point of view, the presented studies exemplify a number of research design approaches to eliminate possible confounding factors. For instance, the same informational content may be delivered across conditions to cleanly separate the effects of non-adaptive aspects ("static" message content) and adaptive aspects (informed selection of messages) (Chaudhuri et al. 2008; Walker et al. 2011). Or a (secondary) control condition with randomly selected messages can be used to determine whether the specific message selection approach is better than random selection (Walker et al. 2011b). Another problem is that adaptation strategies depend on diagnostics that must be processed beforehand. Yet, diagnostics are often imperfect, introducing another confounding factor when trying to evaluate adaptation strategies independently. A possible approach is to resort to a Wizard-of-Oz setup. That is, a human may emulate computational decision making, harnessing human intelligence and judgment to make decisions in a more precise (in the best case, near-perfect) way. Of course, the kind of decision making needed must be manageable for humans in real time. If the adaptation strategy turns out to be ineffective, system developers have saved the effort to develop computational diagnostic modules that would have been needed to realize the strategy. As discussed below, such efforts can amount to considerable work. In general, to facilitate and accelerate future research efforts, tools like the here presented CASE engine, with its ability to conveniently configure alternative support strategies, can be utilized as research platforms. To support the just described Wizard-of-Oz setup, a small extension to the CASE engine would be a wizard interface that allows humans to feed their judgments into the system and make them available to the adaptation machinery.

8.2.2 Automated analysis

The automated adaptation within argumentation learning systems is driven by a more or less sophisticated analysis of students' learning activities. There are two principal approaches to developing analysis systems, the knowledge-driven approach (i.e., handcrafted models of expert knowledge) and the data-driven approach (i.e., models computationally induced from data), both of which discussed in this section.

Subsection 8.2.2.1 sketches the general development processes for adaptive support based on the CASE framework (see Chapter 5), indicating where the development of automated analysis functions ties into the overall development process. Since the design of the CASE framework has been informed through an extensive analysis of adaptive argumentation systems, the described process can be seen as a blueprint for developing such systems more generally, even without the CASE framework. The next two subsections discuss the pros and cons of the knowledge-driven approach (subsection 8.2.2.2) and the data-driven approach (subsection 8.2.2.3). Against this background, subsection 8.2.2.4 compares the knowledge-driven and the data-driven approach, elaborates on possibilities to combine both approaches, and discusses the role structured interfaces may play in facilitating and enabling automated analyses. Main points of this section are summarized in Table 42, p. 271.

8.2.2.1 Place within the overall development process

The development of automated analysis functionality is one step within the development of an adaptive support approach. The corresponding CASE development process typically comprises the following steps:

1. conceptualization of the general support goals and approach, addressing, among other things, the following questions:

   - What aspects of the learning process should be supported (e.g., problem solving, collaboration)?
   - What should be the general nature of the support (e.g., reflective or remedial feedback regarding past activities, forward-looking hints to overcome impasses)?
   - What is the envisioned overall support process (e.g., are there different phases that require different kinds of support)?
2. definition of concrete pattern instances that represent opportunities to provide support
3. definition of feedback messages to be delivered in response to detected patterns
4. definition of the overarching support strategy (when to provide which message)

The development process requires test-revision cycles at different stages to evaluate and improve both technical aspects (functioning of individual modules, interplay of different modules) and the pedagogical effectiveness of the entire setup. In step 2, the CASE framework allows system developers to add pattern detection functionality that may be based either on a knowledge-driven or a data-driven approach.

8.2.2.2 Knowledge-driven analysis

Knowledge-driven approaches involve a process typically referred to as knowledge engineering. In a first step, relevant patterns must be defined on a conceptual level. In a second step, the operational logic to identify these patterns must be specified. As discussed before, the CASE engine utilizes production rule technology to implement a general framework for pattern search in argument diagrams (see Chapter 5). To operationalize pattern search, system developers can either use the CASE Feedback Authoring Tool or directly program corresponding production rules into the system. While the latter option provides more flexibility and freedom in defining patterns, it is also technically more demanding, requiring expertise in expert system programming. System developers may be experts in a given domain and thus capable of identifying and operationalizing target pattern independently, without external input. Alternatively, system developers may focus on the technical aspects and consult external experts to elicit the domain and pedagogical knowledge needed to operationalize the analysis.

Table 42
Analysis of approaches to automated modeling in argumentation learning systems

Approach	Benefits	Limitations
Knowledge-driven models (based on knowledge engineering techniques, e.g., expert system modeling)	**educational benefits** – *(enables automated adaptation with its benefits, see Table 41, p. 249, educational benefits)* **implementation benefits** – manageable (and possibly preferable) approach for patterns of limited complexity – incremental and iterative development of expert knowledge base possible (adding new patterns, improving existing patterns) based on practical experiences and evaluation results – human understandable models and human-predictable model output – current research on the authoring of intelligent and adaptive technologies (to facilitate development process)	**educational limitations** – *(enables automated adaptation with its limitations, see Table 41, p. 249, educational limitations)* **implementation limitations** – knowledge engineering expertise required – detailed domain and pedagogical knowledge required – medium to high costs of knowledge elicitation and modeling – only applicable if required diagnostics are amenable to the knowledge engineering approach (e.g., patterns that are well-defined and not overly complex)
Data-driven models (based on data analysis techniques, e.g., machine learning model induction)	**educational benefits** – *(enables automated adaptation with its benefits, see Table 41, p. 249, educational benefits)* **implementation benefits** – applicable to complex, to some extent ill-defined patterns – iterative, data-driven improvement of diagnostics possible (e.g., through relevance feedback mechanisms) – availability of linguistic technologies and resources (part-of-speech parser, functional dependencies parsers, named-entity recognition, etc.) – current research about domain adaptivity and transfer of machine learned models may help to transfer models to related settings with reasonable implementation efforts and demand for data	**educational limitations** – *(enables automated adaptation with its limitations, see Table 41, p. 249, educational limitations)* **implementation limitations** – machine learning engineering expertise required – sufficient amount of data required, which, in addition, must be labeled, depending on the approach – high costs of data collection, labeling, and machine learning experimentation – risk of negative validation result and wasted development resources if models turn out to have only insufficient predictive power – risk of performance loss when training and target scenario are different – often models not human understandable and model output hard for humans to predict – depending on model complexity, model may need considerable computational resources

The step from the ideation of relevant patterns to the concrete operational definition is typically not trivial. First, there might be problems in eliciting the required knowledge. Fully and precisely capturing the knowledge of human experts is often difficult, in particular, when it comes to tacit knowledge—knowledge that humans are not able to articulate or even aware of (Dahlbom and Mathiassen 1993). Second, the operationalization itself may be troublesome. Intuitively clear pattern descriptions may correspond to complex logical specifications. Thus, the approach is only suitable for patterns of low to moderate complexity; otherwise the manual specification becomes unmanageable. On the positive side, the knowledge engineering approach gives developers complete control over the way how patterns are operationalized. The defined patterns are human-understandable and can be adjusted based on practical experience, e.g., when the detected pattern instances turn out to not quite match the pattern one originally had in mind. In general, the knowledge base of patterns can be incrementally developed and iteratively refined based on informal testing and evaluation. Authoring tools, such as the CASE Feedback Authoring Tool (see section 5.8), can support the development process and enable technical novices to actively participate in that process. In summary, knowledge-driven approaches are particularly appropriate for application domains that are well-understood, with explicit, workable knowledge available, and patterns of low to moderate complexity. Correspondingly, knowledge-driven approaches have been widely used to support knowledge representation activities based on argument diagrams and other formats that provide explicit, machine understandable semantics. Examples are Belvedere (Suthers et al. 2001), Rashi (Woolf et al. 2003), and LARGO (Pinkwart et al. 2009).

8.2.2.3 Data-driven analysis

Data-driven approaches utilize data analysis techniques, such as methods from the fields of machine learning and information retrieval, to computationally infer analysis parameters or complete analysis models. In context of the Argunaut project (see Chapter 6), a supervised machine learning approach was used to induce classifiers for important qualitative aspects of e-discussions from annotated data, which should serve as an example for data-driven approaches here. The process essentially unfolded into two parts: the generation of a data corpus with positive and negative examples for target categories (the machine learning input), and the actual machine learning engineering process (the machine learning analysis). To generate the data corpus, pedagogical experts first defined and operationalized relevant target categories. In contrast to the knowledge-driven approach, the operationalization does not have to be at a level that is interpretable by a computer program. Rather, it only needs to be sufficiently detailed for humans to analyze data in an objective and reliable way. The rationale is that machine learning can later implicitly capture the human intelligence that goes into the analysis to build machine-executable models. Instructions how to apply categories to data segments have been collected in a coding handbook and employed to analyze existing data. The result was a corpus of positive and negative examples for each target category. The machine learning engineering process consisted of a series of experiments with varying attribute combinations (i.e., possible predictor variables that can be computed in relative straightforward ways) and machine learning algorithms. Linguistic technologies have been used to process natural language text and determine language-based attributes. The resultant machine learned models have been cross-validated and selected for practical use if the predictive performance has met a predefined criterion.

In contrast to the knowledge-driven approach, supervised machine learning is also applicable when patterns are complex and their manual specification out of reach. For instance, machine learning can help to induce complex predictive models based on terms, term combinations, and grammatical constellations in natural language texts, which would be hard to manually define by human experts. Therefore, a number of analysis systems for student discussions are based on machine learning (e.g., Rosé et al. 2008; Walker et al. 2011; Argunaut work of this dissertation reported in Chapter 6).

Yet, to apply machine learning, a sufficient amount of data must be available. Particularly supervised machine learning approaches are expensive since the data corpus must be manually labeled beforehand. As a rule of thumb, the more complex a target concept, the more training data is required to induce a machine learning model. Several studies suggest that a corpus size of around 1,000 segments (+/- 300) is sufficient to achieve (close to) satisfactory classification accuracy values ($\kappa > .6$) when analyzing qualitative aspects of computer-mediated education discussions (Ai et al. 2010; Dönmez et al. 2005; Argunaut experiments reported in Chapter 6). Of course, this coding effort multiplies with the number of coding categories. Dönmez et al. (2005) provide concrete numbers regarding a multidimensional coding scheme (7 dimensions). The coding was done in three waves; overall more than 17,000 text segments were coded by about 6 coders per wave. In each wave, coder training took about 500 man-hours, the coding itself about 1,200 man-hours (for about 5600 $\approx$ (17,000 / 3) text segments). That is, to produce the 1000 segments suggested above, one may calculate 700 man-hours of work (500 man-hours for the training, about 200 man-hours for the actual coding). On top of that, the machine learning engineering process itself requires considerable time and effort in conceptualizing, planning, preparing, and conducting experiments. The development costs must also be seen in relation to the scope of applicability of induced classifiers. As suggested by results of the Argunaut experiments (see Chapter 6) and other research (Daumé III and Marcu 2006), the more the application setting differs from the training setting (e.g., student population and discussion topic), the more the classifiers may drop in accuracy. So, additional efforts and retraining may be required to transfer existing classifiers to new application domains. Current research investigates how such efforts can be minimized. For instance, Mu et al. (2012) devised a multi-stage classification approach. The classifiers do not operate directly on the (context-specific) terms contained in the data corpus. Rather, specific terms are pre-classified into more general semantic categories, such as *person*, *location*, and *date*. This allows the classifiers to operate on a more general level, decoupled from specific word choices in the training corpus. Of course, if not available, an additional classifier for tagging terms according to higher-level semantic categories must be developed for each specific target context. Moreover, some loss of specific semantics is unavoidable, which can affect the classification performance both positively (if the removed specifics are unnecessary noise) and negatively (if the lost semantics are relevant with respect to the target categories). Depending on the specific coding categories and data, there might be a need to develop automated data segmentation procedures in additional in order to apply the category classifiers to text segments that constitute the right unit of analysis. Recent research yielded promising results towards automated segmentation in CSCL contexts (Mu et al. 2012).

Another important consideration is the risk of failing in inducing classifiers with sufficient predictive power. There is no guarantee that a machine learning algorithm is able to capture the fine nuances of natural language from a limited set of training data. While human coders can utilize the world knowledge they accumulated over a whole lifetime to make judgments, a machine learning algorithm is, by and large, restricted to a very limited set of available data. The quality and representativeness of this data is crucial to achieve a positive result. So, the number of training examples cited before (around 1,000 examples) may turn out to be insufficient in individual cases. Moreover, even if a classifier surpasses an a priori defined performance threshold (κ = .6, .7, or whatever), this may still not be good enough to achieve an effective adaptation. Depending on the specific application scenario (i.e., how the classifications are finally used), the demands on prediction accuracy may vary. In particular, the often cited and used threshold of κ > .7 was originally defined as a reference point for sufficient agreement between human raters in content analysis (Krippendorff 1980) and not to evaluate adaptive computer technology. So, while such criteria give some orientation and guidance, an ultimate decision is only possible when testing the integrated adaptive support system in the target scenario. If it turns out that classifiers are not accurate enough, developers may already have spent considerable time and resources. Another possible “show-stopper” for practical use is an excessive runtime of classifiers. Possible countermeasures are to focus on machine learning algorithms that produce very efficient classification models (e.g., Support Vector Machines) and to reduce model complexity by keeping down the number of attributes of the final model (e.g., by ruling out attributes that presumably contribute only little). Despite all of the potential problems discussed, some CSCL systems demonstrate that machine learned classifiers can be successfully used to drive adaptations, highlighting the great potential of the approach (Kumar et al. 2007; Walker et al. 2011).

8.2.2.4 Developing automated analysis functions

Knowledge-driven approaches are particularly appropriate for less complex patterns with relatively straightforward semantics in relatively well-understood domains, for instance, to analyze structural constellations in argument diagrams. Yet, the approach becomes intractable for complex patterns, for instance, when natural language is involved. Then, data-driven approaches, such as supervised machine learning, are the methods of choice. Inducing machine learned classifiers from data involves relatively high development costs and the risk that the predictive performance of resultant classifiers is not sufficient for practical application.

Both approaches may be used side by side. For instance, the *Multilevel Scaffolding* scenario described in Chapter 4 may be enhanced with support based on a human-engineered pattern search in the diagrams and machine-learned classifiers in the chat discussions. While most machine learned models are complex, hard to understand, and thus not accessible to manual refinement, they can be combined with knowledge-driven approaches in a staged way. For instance, Ai et al. (2010) developed a component that applies heuristic rules to post-process low-confidence predictions from the machine learned classifiers, achieving a boost in the predictive performance of the overall system. Also the analysis of argument diagrams may profit from combining both approaches. Diagram boxes could be classified based on the contained text (e.g., as unjustified claims and critical comments), and the rule-based approach could search for larger patterns in the diagrams utilizing these classifications (e.g., an unjustified claim that is not responded to with a critical comment). The CASE engine provides the technical infrastructure to implement such scenarios: Rule-based agents and machine-learned agents can share their analysis results through a joint knowledge base.

One of the main challenges in developing automated analysis mechanisms is how to come to grips with natural language user input, which is the most common form to express arguments and therefore of high practical relevance. As discussed before, knowledge-driven approaches are not appropriate to capture semantics of natural language texts at a deep level. While data-driven approaches are more promising in this respect, they suffer from a number of other difficulties: They require a sufficient amount of existing data, are relatively expensive in terms of development costs, do not necessarily generalize well beyond the training scenario, and are always at risk to not achieve sufficiently reliable models. A possible solution to make the analysis of natural language more tractable is to capitalize on structured user input, which can be provided through argument diagrams and sentence openers. Such structures do not only scaffold student reasoning, problem solving, and collaboration through explicit semantics exposed in the user interface. Rather, automated analysis approaches can also utilize these explicit semantics to interpret user inputs. Most approaches to analyze argument modeling activities ignore natural language texts and solely focus on the explicit semantics of the model structure (Pinkwart et al. 2009; Ranney and Schank 1998; Suthers et al. 2001). The Argunaut experiments (Chapter 6) demonstrate how diagram structures can be used in conjunction with natural text input to induce machine learned classifiers from data. Also the analysis of student discussions can benefit from explicit structures. In the past, researchers explored the potential of sentence openers as an easy way to capture the intentions of discussion moves. For instance, McManus and Aiken (1995) used sentence openers to trigger transitions of a state automaton that modeled possible states of a discussion (e.g., *display*, *confirm*, *disconfirm*, and *convergence*). Others used sentence openers as machine learning attributes to induce predictive models of group interaction problems (Goodman et al. 2005; Soller 2004). The next chapter develops these ideas further in the context of the *Multilevel Scaffolding* scenario described in Chapter 4.

Chapter 9

Adaptive Support for the Multilevel Scaffolding Scenario

To bring together the different findings of the dissertation in a concrete way, this chapter sketches how the *Multilevel Scaffolding* scenario described in Chapter 4 could be supported through adaptive support. In particular, this chapter addresses how the structural elements of the user interface—diagram elements and sentence openers—could be exploited to implement an automated analysis of chat discussions. First insights regarding the feasibility and challenges of the approach will be presented based on the results of a pre-study.

As discussed before, effective feedback should consider both content and process aspects of student discussions. To facilitate the analysis on the process level, the sentence openers currently used as part of the FACT-2 script could be utilized to tag messages according to their communicative intentions. To facilitate the analysis on the content level, explicit diagram references could be added to the chat as a second structuring element. That is, if a new chat contribution cites or takes position towards elements of the diagram, students would explicitly indicate which diagram element the chat messages is referring to. Diagram elements again represent passages of the texts that students analyze and discuss. Therefore, students could use another reference mechanism to indicate which text passages diagram elements are based upon. The topic of chat contributions could then be determined by (1) identifying the diagram element referenced in the chat contribution, (2) identifying the text passage referenced in the diagram element, and (3) identifying the topic of the text passages through a predefined model (e.g., each text passages may be annotated with one or multiple topic tags). Building upon well-structured and machine interpretable user input, the described framework may allow analyzing student discussions in a more accurate way on both the process and content dimension.

These two pieces of information could be used independently to support student discussions. Students' use of sentence openers gives some indication of their discussion style, e.g., whether they take a critical stance (indicated by a large amount of *critic* sentence openers) or not (indicated by a small amount of *critic* sentence openers). Students who rarely criticize or object to messages of their fellow students could be prompted to be more critical. Students' references to the diagram give some indication of the contents that have been covered so far and the extent to which these topics have been covered. Based on this information, the system could prompt students to cover important topics not yet discussed, to elaborate more on important yet under-discussed points, or to speed up their discussion when time is running short and many important aspects have not been addressed yet. This might be useful since scripted discussions put an additional load on students, which may result in a reduced number of task aspects covered (Weinberger et al. 2010). Combining content and process aspects may provide an even more powerful framework. For instance, it may be possible to determine how critically specific contents have been discussed and which connections have been made between the two opposing texts (e.g., some data reported in text *A* has been used to attack some claim made in text *B*). In theory, the information derived from the described framework can be exploited in manifold ways to support student discussions. In practical terms, the crucial question is how complete and reliable the derived information actually is. To achieve high-quality inferences it is essential that students make sufficient and consistent use of structuring elements. A feasibility analysis based on the data of the *Multilevel Scaffolding* study investigated possible obstacles to such an approach (Scheuer et al. 2013).

As reported above, in one out of five messages a sentence opener was used, and two thirds of all dyads made frequent or occasional use of sentence openers according to an a priori defined criterion (frequent: more than 25% of all messages use a sentence opener, occasional: between 10% and 25% of all messages use a sentence opener). While sentence opener misuse almost never occurred, there was nevertheless a wide variety of different uses. For instance, the sentence opener *Could you explain to me ...* was not only used to elicit explanations in a neutral way, but also to raise concerns and objections against previous points. The sentence opener *For instance ...* was not only used to illustrate some previous point, but also to list one or more exemplary arguments to support a previous point. In line with the observation of Israel and Aiken (2007), some messages expressed multiple independent ideas at once, another potential problem for an automated analysis. For instance, some students present an argument and ask a question within the same message. In this case, the sentence opener only identifies the first part of the message (the argument) but not the second part (the question). In other cases, students provide several independent reasons to support a claim in the same message. A computational model might then assume that only one reason was given. Finally, the point of reference of messages is not always clear. For instance, the sentence opener *A supporting argument is ...* is sometimes used to present an argument regarding a recent claim (which might or might not be presented in the latest message). In other cases, the message refers to the general discussion topic. In either case, supporting sentence openers are typically used to support one's own position and opposing sentence openers to oppose the partner's position, a useful heuristic to interpret sentence openers.

The analysis revealed also that diagram references were used at a reasonable rate. Across all dyads about 12% of all messages contained a diagram reference. Again, the extent of usage differed considerable between dyads: five dyads made frequent use (> 25% diagram referencing messages), five dyads made occasional use (> 10%), and twelve dyads made rare use of diagram references (< 10%). That is, almost half of all dyads (45%) used diagram references in their contributions at least occasionally. A rate of 10%, or even 25%, is quite substantial, since about one third of all messages are not about elaborating on the subject matter. Moreover, it is certainly not unusual that students exchange multiple messages regarding one and the same diagram element. Also, the use of diagram references was not mandatory; students were only hinted at the possibility of doing so. There were also a variety of ways diagram references were used, which is certainly an artifact of the lack of clear instructions, and a potential problem in an automated analysis of diagram references. For instance, diagram references were added to messages to cite the diagram element as the source of the information used in the message. In other cases, diagram references were used to comment on the content of a diagram element. In other cases again, students used diagram references as a shortcut or placeholder for the content of the referenced element, to save the effort of typing the complete statement into the chat. In one instance, a student explicitly complained about this practice, annoyed with searching for the referenced contents in the diagram. Finally, students had different approaches in where to post diagram references. Some students used diagram references in the very message it belongs to, others posted the diagram reference in a separate message afterwards, and others first posted the diagram reference and then in a second message included the actual content of the referenced diagram element.

In summary, and in contrast to some previous results (e.g., Lazonder et al. 2003), most students made reasonable use of sentence openers and diagram references. Whether the extent of use provides sufficient information to effectively support students is an empirical question still to be investigated. Even if students make frequent use of the provided structures, they may appropriate them in different ways, giving rise to multiple possible interpretations, a possible obstacle to generate precise feedback. Some level of uncertainty might not be a problem depending on the specific objectives one is pursuing. For instance, there is no need to resolve the exact point of reference of messages if one abstains from a fine-grained computational model. Rather than capturing the amount of support and opposition for specific statements, such a model could focus on the number of pros and cons regarding the two main positions. Of course, more coarse-grained assessments also limit the possibilities for feedback generation. Another approach is to use simple computational heuristics to disambiguate well-structured input. For instance, Israel and Aiken (2007) used keyword matching in addition to sentence openers to classify messages. Another option is to use the information gained through structuring interface elements to enhance a machine learning analysis. For instance, machine-learned classifiers may determine the communicative intention of messages based on sentence openers and natural language text, an approach similar to the Argunaut approach described in Chapter 6. Recent research investigates machine learning approaches to determine the response structure of chats (e.g., Mayfield et al. 2012), another possible problem observed in the pre-study.

The learning environment itself could also be enhanced to guide students towards a more effective and uniform usage of structuring elements. In particular the following enhancements may be employed:

- Message categories (e.g., claim, argument, question) could replace sentence openers since, if chosen well, they have a clearer meaning and leave less room for interpretation. Moreover, they can be used more easily (and might therefore be used more frequently), since they do not require students to fit their messages syntactically to their selection. On the downside, categories do not provide a scaffold at the level of the concrete way of formulating a specific type of statement.
- Discussion threads could disambiguate the point of reference of messages since each message then has a clearly defined predecessor. Alternatively, students could explicitly indicate to which previous chat message a new message refers to. This approach is used, for instance, in the ConcertChat tool (Mühlpfordt and Wessner 2005). On the downside, this kind of structuring imposes an additional burden on the user, who would have to create these structures.

- More explicit instructions could make expectations and recommended forms of usage clearer (e.g., appropriate message grain-size; how to use diagram references).
- Incentives could encourage more frequent use of sentence openers and diagram references. For instance, the system could highlight referenced diagram elements when students hover with the mouse pointer over chat messages, thus, save them from a time-consuming search in the diagram, a clear value added. As a social incentive, a score based on the number of sentence openers each student used could be displayed. On the downside, such social, game-like incentives may provoke behaviors such as *gaming-the-system* (Baker et al. 2004). That is, students may inappropriately use sentence openers to acquire higher scores rather than to engage in a higher-quality discussion.
- The creation of diagram references could be simplified, e.g., diagram references could be added to messages through a drag-and-drop functionality.
- Explicit feedback could specifically target an appropriate use of structuring elements, e.g., prompting messages for students who make only rare use of sentence openers. On the downside, as discussed before, there is also a cost to it since each intervention on the part of the system potentially interferes with natural student-to-student interactions.

It should not be forgotten that the generation of effective feedback is only one concern in the design of adaptive argumentation learning systems and sometimes at odds with usability and pedagogical concerns, as indicated by the drawbacks mentioned in the list above. That is, improving the accuracy of an automated analysis might have undesirable side effects. Therefore, one has to carefully consider the tradeoffs involved in the design of argumentation learning user interfaces and scripts.

In summary, highly structured user interfaces have the potential to guide students towards modes of interaction closer to the ideal model of interaction the instructional designer had in mind. The constraints and affordances incorporated into the user interface give the system designer some level of control over the kind of interactions that are possible or expectable—important theories (Fischer et al. 2013; Suthers 2003) and empirical research regarding the aspect of guidance induced through user interfaces have been extensively discussed throughout this dissertation. Moreover, as discussed in this chapter, highly structured user interfaces provide better-structured user inputs, which can inform the automated analysis of interactions in a more precise way. On the negative side, there is the danger that an interface may become too restrictive, which then may lead to mechanical and unnatural forms of interactions—a potential betrayal of important ideals of computer-supported collaboration (Weinberger 2011), such as promoting rich and authentic forms of interactions. Moreover, the same theories that highlight the potential of well-designed structured interfaces also point at negative consequences that may arise. For instance, collaboration scripts may obstruct fruitful forms of interaction unanticipated by the designer. As in the case of sentence openers, only a relatively small number of moves can be represented to not overload the user interface (Soller 2001). Knowledge representations may *mis*-guide students' attention, as evidenced by excessive prompting exerted by a table notion in a study by Suthers and Hundhausen (2003). Moreover, knowledge representation activities involve an inherent overhead since users must learn how to use corresponding notational systems and spend time and effort keeping the representation well-organized and readable (Buckingham Shum et al. 1997). The increased control by the designer mentioned above may, conversely, lead to user frustration and decreased engagement, in particular, when students feel patronized and perceive the structure as a burden rather than an aid. All in all, the tradeoffs involved in the design of computer-based argumentation learning systems, both, adaptive and non-adaptive, highlight the need for sustained research efforts in this area.

Chapter 10
Conclusions

This dissertation investigated crucial aspects in the design of argumentation learning systems. Its first part focused on the question of how user interfaces could be designed to promote high-quality actions and *inter*-actions of learners. Empirical results showed that discussion scripts can be employed to significantly raise the amount of elaboration that occurs during educational discussions. Besides increasing the amount of high-quality moves, scripts generally help learners keep a focus on the learning task, as indicated by previous research (Baker and Lund 1997; Oh and Jonassen 2007) and a significantly lower proportion of moves at a low elaboration level in the here presented study.

The mechanism at work may be explained based on the script theory of guidance (Fischer et al. 2013). Discussion scripts are composed of structuring elements (e.g., sentence openers) that help learners activate knowledge about discussion and collaboration practices they often already possess—for instance, presenting a claim, formulating a counterargument, or backing an argument with evidence. Sustained engagement in so scaffolded activities is assumed to help learners develop higher-level mental organizational structures of knowledge, which are then available in the future. In other words, sustained engagement in scripted discussion practice is assumed to promote the acquisition of argumentation skills. Through the FACT-2 script of this dissertation, learners may internalize the concept of a critical discussion, which comprises being both a proponent and a critic. According to the script theory, they can accomplish this by developing knowledge structures regarding the behavior of proponents and critics (role-level script components) based on knowledge about the production of specific individual discussion moves they already possess (scriptlet-level script components). This process is guided by the sentence openers in the user interface, which indicate typical and allowable discussion moves for each role. Generally, while providing a helpful scaffold for novice learners, the application of scripting at a fine-grained interaction level, such as the sentence openers used here, may turn out to be a hindrance, rather than a help, for experienced discussants.

Important questions for future research include whether improvements achieved in scripted discussions actually transfer to unsupported discussions and under which conditions (e.g., how many training sessions are needed), whether discussion scripts also promote the acquisition of domain knowledge and domain-specific reasoning skills (current results are mixed), and under which specific conditions discussion scripts support or hinder learning (for instance, as mentioned above, the mastery level of learners may be an important factor). From a learning-theoretical perspective, a crucial goal must be to better understand, in general terms, how collaboration scripts influence a learner's thinking and acting. The recently proposed script theory of guidance (Fischer et al. 2013) provides a promising starting point, based on a thorough review of existing scripting results, yet, with a number of important questions still open. For instance, Fischer et al. (2013) note that their proposed conceptual framework with the four levels play, scene, role, and scriptlet may be extended, refined, or revised based on new empirical results.

Another area for future research is to investigate how different structuring elements can be combined to increase the effectiveness of instructional approaches (Tabak 2004). The learning environment used in this study is an instance of such a combined approach, utilizing both a discussion script and argument diagrams. It is well documented in previous research that argument diagramming in itself is an effective approach to teach reasoning and thinking skills, and to provide guidance to discussions. Yet, an additional discussion script may help learners better capitalize on the knowledge and reasoning benefits induced by diagrams. Vice versa, because diagrams make information more accessible and facilitate reasoning, they may encourage and support learners in using high-quality discussion moves included in a script. The research design employed in the reported study does not allow definite conclusions on this interesting question, which is up to future research.

Such research could also shed light on a somewhat underrepresented aspect of the script theory of guidance. Currently, the relationship between scripts and knowledge representations is somewhat loose—external knowledge representations are described as elements of the environment that provide situational constraints and affordances that lead learners to select and employ specific components of their mental scripts. Still open is the question of how exactly external knowledge representations influence the dynamic configuration of mental script components. Moreover, the script theory does not address the question of how the combination of external knowledge representations and scripts influences the dynamic configuration and development of mental scripts. A straightforward method to investigate the first question—the effects knowledge representations have on mental scripts—would be to determine whether particular sentence openers are used at an increased rate when specific representational notations are employed. This would allow for drawing conclusions regarding the mental script components activated through specific properties of representational notations. For instance, if a notation includes a *data* element, learners may increasingly use sentence openers for backing up arguments with evidence (e.g., *According to a statistic ...*). Such a result would suggest that an explicit representation of *data* in a notational system activates corresponding scriptlets in the learner's internal repertoire of moves. Of course, this methodology only works out when effects of the notational system are not obscured by the external script, a possibility that leads to the second question—the effects combinations of representational notations and external scripts have on mental scripts. To investigate this second question, a qualitative analysis of chat protocols would be required. Intuitively, one would expect that the use of specific discussion moves increases with the amount of corresponding structural support: (openers + diagrams) > (openers alone) = (diagrams alone) > (no support). A positive result would underline the possible merits of combining different scaffolds. An even more ambitious hypothesis would postulate a synergistic interplay of both structuring elements, which should result in a statistical interaction effect.

The second part of this dissertation focused on several aspects regarding adaptive support of argumentation learning. The presented CASE engine provides a proof-of-concept of how the technological infrastructure for adaptive argumentation learning systems may be designed. It addresses several important concerns, most notably, the ability to flexibly support argumentation across different domains and application scenarios. This objective has been achieved through a comprehensive configuration framework, designed based on adaptive support approaches used in past argumentation learning systems, and an open architecture with predefined extension interfaces to plug in new functionality. To demonstrate the flexibility of the approach, support mechanisms for several scenarios have been implemented, including the analysis of legal transcripts, the planning and preparing of scientific essays, group deliberation about science and math problems, and the moderation of discussions about controversial topics. While a flexible technological infrastructure for adaptive support is important, the perhaps more important question is how to exactly employ adaptive support to promote argumentation learning. Until now, this important issue has been insufficiently addressed. Rigorous empirical testing of the effectiveness of adaptive support in CSCL systems are still the exception rather than the rule—some of these exceptions are discussed in this dissertation (e.g., Kumar et al. 2007; Walker et al. 2011). Similar to the script theory of guidance and the theory of representational guidance, it would be important to develop a unifying theory of adaptive guidance, a theory from which guiding principles for designing adaptive CSCL argumentation systems could be derived. It can be expected that such a theory diverges in important ways from what is known about adaptation strategies to support individual learners, e.g., in traditional intelligent tutoring systems. A unique characteristic of CSCL is that the interactions between co-learners—rather than the interactions between a learner and a system—take the primary role. This particularity of CSCL requires a rethinking of how adaptive support can be delivered, aiming at approaches that do not interfere with natural interactions between learners. To build such a theory, focused research is needed to isolate the effects of individual parameters of adaptive support, e.g., timing, general presentation format, or wording, if textual messages are used. Such a theory may provide an important complement to the script theory of guidance by covering aspects of adaptive CSCL scripts. For instance, an emerging topic—sometimes discussed but still largely unresearched—is the automated fading of scaffolds (Pea 2004), an approach to gradually hand over competencies from the system to the learners.

Adaptations should be tailored to the collaboration and argumentation processes learners engage in. Therefore, an automated and accurate analysis of key aspects of these processes is an important prerequisite for realizing effective adaptation strategies. This dissertation investigated two principal approaches to the automated analysis of argumentation learning activities. As part of the CASE engine, a knowledge-driven approach was realized, which allows defining patterns in argument diagrams that represent aspects of problem solving and collaboration (e.g., modeling errors and problematic interaction patterns). To facilitate the process of defining such patterns, a novel authoring tool was conceptualized and developed. A major contribution is a graphical language to specify patterns based on a visual language. A first usability evaluation yielded encouraging results. Generally, knowledge-driven approaches are particularly suitable when patterns are well-understood (i.e., detailed expert knowledge is available), not too complex (i.e., manual specification is a manageable task), and easily expressible in terms of the available input data (e.g., diagrams typically provide semantically rich input data). The second approach—data-driven induction of relevant patterns—can be employed when a knowledge-driven realization of the analysis is out of reach, e.g., when an analysis of natural language is a key to success. This dissertation presented the first machine learning approach to automatically inducing classifiers for qualitative aspects of graphical discussions (e.g., off-topic contributions and argument-counterargument interactions), with promising validation results.

One main future challenge with respect to the application of data-driven approaches, such as machine learning, is ensuring a sufficient accuracy of the developed models. Such models typically incorporate regularities detected in training data, which are often, but not always, predictive of some phenomenon of interest. That is, in some cases, misclassifications will occur, which, in turn, may lead to inappropriate adaptations potentially harmful for learning. One option, and potential component of a theory of adaptation, is fault-tolerant adaptation, that is, an approach to adaptation that is not, or only minimally, harmful even in the case of a fallible analysis. A second option is to find ways to improve the accuracy of the analysis itself. A promising way to advance is importing and testing ideas and algorithms from both, foundational research (e.g., basic machine learning and natural language processing research) and related applied fields (e.g., social analytics and user modeling). Another option is to enrich the input data to be analyzed with meaningful information. Structuring elements of the user interface, such as diagrams, sentence openers, and explicit referencing mechanisms, can be exploited to enhance recorded user data with semantically-rich information, which can potentially contribute to a more accurate analysis. Of course, one has to take care that the structuring does not get into the way of natural user interactions and impair learning. A second approach to enrich the input data is to use a staged analysis. Before applying a data-driven approach, a knowledge-driven approach may be employed to identify straightforward, but meaningful, patterns in the input data—an approach particularly attractive when a knowledge-driven approach is already in place. Such an approach makes expert knowledge accessible to a data-driven analysis that would otherwise not be available. The stages can also be switched: First, hard-to-formalize patterns may be identified based on a data-driven approach (e.g., machine learning). Then, specific constellations of these patterns, relevant for driving the adaptation of a system, can be manually defined. In this case, the data-driven analysis can potentially reduce the complexity of the input data to a level manageable by a human expert. Overall, research on such combinations of knowledge and data-driven analysis approaches is still rare and a promising avenue for future research.

References

Ai, H., Sionti, M., Wang, Y. C., & Rosé, C. P. (2010). Finding transactive contributions in whole group classroom discussions. In K. Gomez, L. Lyons, & J. Radinsky (Eds.), *Learning in the disciplines: Proceedings of the 9th International Conference of the Learning Sciences – Volume 1, Full papers* (pp. 976–983). Chicago, IL, USA: International Society of the Learning Sciences.

Ainsworth, S. E. (1999). *Designing effective multi-representational learning environments.* Technical Report number 58. ESRC Centre for Research in Development, Instruction and Training, University of Nottingham, U.K.

Aleven, V., McLaren, B., Roll, I., & Koedinger, K. (2006). Toward meta-cognitive tutoring: A model of help seeking with a Cognitive Tutor. *International Journal of Artificial Intelligence in Education*, *16*(2), 101–128.

Anderson, J. R., Corbett, A. T., Koedinger, K., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences, 4*(2), 167–207.

Anderson, R. C., Nguyen-Jahiel, K., McNurlen, B., Archodidou, A., Kim, S., Reznitskaya, A., Tillmanns, M., & Gilbert, L. (2001). The snowball phenomenon: Spread of ways of talking and ways of thinking across groups of children. *Cognition and Instruction, 19*(1), 1–46.

Andriessen, J. Baker, M., & Suthers, D. (Eds.). (2003). *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments*. Dordrecht: Kluwer.

Andriessen, J. (2006). Arguing to learn. In K. Sawyer (Ed.), *Handbook of the learning sciences* (pp. 443–459). Cambridge University Press.

Andriessen, J., & Schwarz, B. (2009). Argumentative design. In N. Muller Mirza, & A.-N. Perret-Clermont (Eds.), *Argumentation and education: Theoretical foundations and practices* (pp. 145–174). New York: Springer.

Ashley, K. (1990). *Modeling legal argument: Reasoning with cases and hypotheticals*. Cambridge, MA: MIT Press/Bradford Books.

Asterhan, C. S. C., & Schwarz, B. B. (2007). The effects of monological and dialogical argumentation on concept learning in evolutionary theory. *Journal of Educational Psychology, 99*(3), 626–639.

Asterhan, C. S. C., Wichmann, A., Mansour, N., Wegerif, R., Hever, R., Schwarz, B. B., & Williams, M. (2008). *Argunaut deliverable D6.3: Evaluation report on the pedagogical content of the Argunaut system (revised)*. Accessible Online. http://www.argunaut.org/ARGUNAUT_-_D6-3.zip

Asterhan, C. S. C., & Schwarz, B. B. (2009). The role of argumentation and explanation in conceptual change: Indications from protocol analyses of peer-to-peer dialogue. *Cognitive Science, 33*(3), 373–400.

Austin, J. L. (1962). *How to do things with words*. Oxford: Clarendon Press.

Azevedo, R., Johnson, A., Chauncey, A., & Burkett, C. (2010). Self-regulated learning with MetaTutor: Advancing the science of learning with MetaCognitive tools. In M. S. Khine & I. M. Saleh (Eds.), *New science of learning* (pp. 225–247). New York: Springer.

Azevedo, R., Cromley, J. G., Moos, D. C., Greene, J. A., & Winters, F. I. (2011). Adaptive content and processing scaffold: A key to facilitating students' self-regulated learning with hypermedia. *Psychological Test and Assessment Modeling, 53*(1), 106–140.

Baghaei, N., Mitrovic, A., & Irwin, W. (2007). Supporting collaborative learning and problem solving in a constraint-based CSCL environment for UML class diagrams. *International Journal of Computer-Supported Collaborative Learning, 2*(2–3), 159–190.

Baker, M. J., & Lund, K. (1997). Promoting reflective interactions in a computer-supported collaborative learning environment. *Journal of Computer Assisted Learning*, *13*, 175–193.

Baker, M. J. (2003). Computer-mediated argumentative interactions for the co-elaboration of scientific notions. In J. Andriessen, M. J. Baker & D. D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments* (pp. 47–78). Dordrecht, the Netherlands: Kluwer Academic Publishers.

Baker, M., Andriessen, J., Lund, K., van Amelsvoort, M., & Quignard, M. (2007). Rainbow: A framework for analyzing computer-mediated pedagogical debates. *International Journal of Computer-Supported Collaborative Learning*, *2*(2–3), 247–272.

Baker, R. S., Corbett, A. T., Koedinger, K. R., & Wagner A. Z. (2004). Off-task behavior in the cognitive tutor classroom: when students game the system. In E. Dykstra-Erickson, & M. Tscheligi (Eds.), *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 383–390). New York: ACM Press.

Barth, E. M., & Krabbe, E. C. W. (1982). *From axiom to dialogue. A philosophical study of logics and argumentation.* Berlin / New York: Walter de Gruyter.

Bell, P. (1997). Using argument representations to make thinking visible for individuals and groups. In R. Hall, N. Miyake, & N. Enyedy (Eds.), *Proceedings of the 2nd International Conference on Computer Support for Collaborative Learning* (pp. 10–19). Toronto, Ontario: University of Toronto Press.

Ben-David, A. (2006). What's wrong with hit ratio? *IEEE Intelligent Systems, 21*(6), 68–70.

Berkowitz, M. W., & Gibbs, J. C. (1983). Measuring the developmental features of moral discussion. *Merrill-Palmer Quarterly, 29*, 399–410.

Berkowitz, M. W., & Gibbs, J. C. (1985). The process of moral conflict resolution and moral development. In M. W. Berkowitz (Ed.), *Peer conflict and psychological growth* (pp. 71–84). San Francisco, CA: Jossey-Bass.

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher, 13*(6), 4–16.

Bodemer, D., & Dehler, J. (2011), Group awareness in CSCL environments. *Computers in Human Behavior, 27*(3), 1043–1045.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In D. Haussler (Ed.), *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory* (pp. 144–152). New York: ACM Press.

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher, 18*(1), 32–42.

Brown, A. L. (1992). Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings. *The Journal of the Learning Sciences, 2*(2), 141–178.

Brown, D. A. (2002). U.S. obligation to act even if the developing world does not. In: D. A. Brown, *American heat: Ethical problems with the United States' response to global warming*. Rowman & Littlefield, New York.

Buchs, C., Pulfrey, C., Gabarrot, F., & Butera, F. (2010). Competitive conflict regulation and informational dependence in peer learning. *European Journal of Social Psychology, 40*(3), 418–435.

Buckingham Shum, S. J., MacLean, A., Bellotti, V. M. E., & Hammond, N. V. (1997). Graphical argumentation and design cognition. *Human-Computer Interaction, 12*(3), 267–300.

Buder, J., & Bodemer, D. (2008). Supporting controversial CSCL discussions with augmented group awareness tools. *International Journal of Computer-Supported Collaborative Learning, 3*(2), 123–139.

Bumiller, E. (2005, August 3). Bush remarks roil debate on teaching of evolution. *The New York Times*. Retrieved from http://www.nytimes.com/2005/08/03/politics/03bush.html

Byrne, R. M. J. (1989). Suppressing valid inferences with conditionals. *Cognition, 31*(1), 61–83.

Cavalier, R., & Weber, K. (2002). Learning, Media, and the Case of Dax Cowart: A Comparison of Text, Film, and Interactive Media. *Interactive Learning Environments, 10*(3), 242–262.

Chaudhuri, S., Kumar, R., Joshi, M., Terrell, E., Higgs, F., Aleven, V., & Rosé, C. P. (2008). It's not easy being green: Supporting collaborative "green design" learning. In B. P. Woolf, E. Aimeur, R. Nkambou, & S. Lajoie (Eds.), *Proceedings of the 9th International Conference on Intelligent Tutoring Systems* (pp. 807–809), Berlin: Springer.

Chaudhuri, S., Kumar, R., Howley, I., & Rosé, C. P. (2009). Engaging collaborative learners with helping agents. In V. Dimitrova, R. Mizoguchi, B. du Boulay, & A. Graesser (Eds.), *Proceedings of the 14th International Conference on Artificial Intelligence in Education* (pp. 365–372). Amsterdam: IOS Press.

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13*(2), 145–182.

Chi, M. T. H., de Leeuw, N., Chiu, M. H., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science, 18*(3), 439–477.

Chi, M. T. H., Siler, S. A., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science, 25*(4), 471–533.

Chi, M. T. H., Roy, M., & Hausmann, R. G. M. (2008). Observing tutorial dialogues collaboratively: Insights about human tutoring effectiveness from vicarious learning. *Cognitive Science, 32*(2), 301–341.

Chi, M. T. H. (2009). Active-constructive-interactive: A conceptual framework for differentiating learning activities. *Topics in Cognitive Science, 1*(1), 73–105.

Chinn, C. A., & Clark, D. B. (2013). Learning through collaborative argumentation. In C. E. Hmelo-Silver, C. A. Chinn, & A. M. O'Donnell (Eds.), *The international handbook of collaborative learning* (pp. 314-332). New York: Routledge.

Clark, D. B., Sampson, V., Weinberger, A., & Erkens, G. (2007). Analytic frameworks for assessing dialogic argumentation in online learning environments. *Educational Psychology Review, 19*(3), 343–374.

Clark, D. B., D'Angelo, C. M., & Menekse, M. (2009). Initial structuring of online discussions to improve learning and argumentation: Incorporating students' own explanations as seed comments versus an augmented-preset approach to seeding discussions. *Journal of Science Education and Technology, 18*(4), 321–333.

Clark, H. H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science, 13*(2), 259–294.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46.

Cohen, E. G. (1994). Restructuring the classroom: Conditions for productive small groups. *Review of Educational Research, 64*(1), 1–35.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453–494). Hillsdale, NJ, England: Lawrence Erlbaum.

Conklin, J., & Begeman, M. L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *ACM Transactions on Information Systems, 6*(4), 303–331.

Constantino-González, M. A., Suthers, D. D., & Escamilla de los Santos, J. G. (2003). Coaching web-based collaborative learning based on problem solution differences and participation. *International Journal of Artificial Intelligence in Education, 13*(2), 263–299.

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction, 4*(4), 253–278.

Corbett, A. T., Koedinger, K. R., & Anderson, J. R. (1997). Intelligent tutoring systems. In M. G. Helander, T. K. Landauer, & P. V. Prabhu (Eds.), *Handbook of human-computer interaction (2nd edition)* (pp. 849–874), New York: Elsevier.

Cuban, L. (1986). *Teachers and machines: The classroom use of technology since 1920*. New York: Teachers College Press.

Dahlbom, B., & Mathiassen, L. (1993). *Computers in context: The philosophy and practice of systems design*. Oxford, U.K.: Blackwell.

Dansereau, D. F. (2005). Node-link mapping principles for visualizing knowledge and information. In S.-O. Tergan, & T. Keller (Eds.), *Knowledge and information visualization, LNCS 3426* (pp. 61–81). Berlin: Springer.

Darnon, C., Doll, S., & Butera, F. (2007). Dealing with a disagreeing partner: Relational and epistemic conflict elaboration. *European Journal of Psychology of Education, 22*(3), 227–242.

Daumé III, H., & Marcu, D. (2006). Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research, 26*(1), 101–126.

De Groot, R., Drachman, R., Hever, R., Schwarz, B. B., Hoppe, U., Harrer, A. De Laat, M., Wegerif, R., McLaren, B. M., & Baurens, B. (2007). Computer-supported moderation of e-discussions: The ARGUNAUT approach. In C. Chinn, G. Erkens, & S. Puntambekar (Eds.), *Computer-Supported Collaborative Learning: Mice, Minds, and Society, Proceedings of the 7th International Computer Supported Collaborative Learning Conference, Part 2* (pp. 165–167), Mahwah, NJ: Lawrence Erlbaum.

Deutsch, M. (2005). Cooperation and conflict: A personal perspective on the history of the social psychological study of conflict resolution. In M. A. West, D. Tjosvold, & K. G. Smith (Eds.), *The essentials of teamworking: International perspectives* (pp. 1–36). Chichester, U.K.: Wiley.

Dillenbourg, P., Baker, M., Blaye, A., & O'Malley, C. (1996). The evolution of research on collaborative learning. In E. Spada & P. Reimann (Eds.), *Learning in humans and machine: Towards an interdisciplinary learning science* (pp. 189–211). Oxford: Elsevier.

Dillenbourg, P. (1999). What do you mean by collaborative learning? In P. Dillenbourg (Ed.), *Collaborative learning: Cognitive and computational approaches* (pp. 1–19). Oxford: Elsevier.

Dillenbourg, P. (2002). Over-scripting CSCL. In P. A. Kirschner (Ed.), *Three worlds of CSCL: Can we support CSCL* (pp. 61–91). Heerlen: Open University of the Netherlands.

Dillenbourg, P., & Hong, F. (2008). The mechanics of CSCL macro scripts. *International Journal of Computer Supported Collaborative Learning, 3*(1), 5–23.

Diziol, D., Walker, E., Rummel, N., & Koedinger, K. (2010). Using intelligent tutor technology to implement adaptive support for student collaboration. *Educational Psychology Review, 22*(1), 89–102.

Doise, W., & Mugny, W. (1984). *The social development of the intellect*. Oxford: Pergamon Press.

D’Mello, S., Olney, A., & Person, N. (2010). Mining collaborative patterns in tutorial dialogues. *Journal of Educational Data Mining, 2*(1), 2–37.

D'Mello, S., & Graesser, A. (2012). AutoTutor and Affective AutoTutor: Learning by talking with cognitively and emotionally intelligent computers that talk back. *ACM Transactions on Interactive Intelligent Systems, 2*(4), 23:2–23:39.

Dönmez, P., Rosé, C. P., Stegmann, K., Weinberger, A., & Fischer, F. (2005). Supporting CSCL with automatic corpus analysis technology. In T. Koschmann, D. Suthers, & T. W. Chan (Eds.), *Proceedings of the Conference on Computer Supported Collaborative Learning* (pp. 125–134), Mahwah, NJ: Lawrence Erlbaum.

Dragon, T., Woolf, B. P., Marshall, D., & Murray, T. (2006). Coaching within a domain independent inquiry environment. In M. Ikeda, K. D. Ashley, & T.-W. Chan (Eds.), *Proceedings of the 8th International Conference on Intelligent Tutoring Systems, LNCS 4053* (pp. 144–153). Berlin: Springer.

Dragon, T., Mavrikis, M., McLaren, B. M., Harrer, A., Kynigos, C., Wegerif, R., & Yang, Y. (2013). Metafora: A web-based platform for learning to learn together in science and mathematics. *IEEE Transactions on Learning Technologies, 6*(3), 197–207.

Driver, R., Newton, P., & Osborne, J. (2000). Establishing the norms of scientific argumentation in classrooms. *Science Education, 84*(3), 287–312.

Dwyer, C. P., Hogan, M. J., & Stewart, I. (2013). An examination of the effects of argument mapping on students' memory and comprehension performance. *Thinking Skills and Creativity, 8*, 11–24.

Dyke, G., Adamson, A., Howley, I., & Rosé, C. (2012). Towards academically productive talk Supported by conversational agents. In S. A. Cerri, W. J. Clancey, G. Papadourakis, & K. Panourgia (Eds.), *Proceedings of the 11th International Conference on Intelligent Tutoring Systems, LNCS 7315* (pp. 531–540). Springer, Berlin.

Dyke, G., Adamson, D., Howley, I., & Rosé, C. P. (2013). Enhancing scientific reasoning and discussion with conversational agents. *IEEE Transactions on Learning Technologies, 6*(3), 240–247.

Dzikovska, M., Steinhauser, N., Farrow, E., Moore, J., & Campbell, G. (2014). BEETLE II: Deep natural language understanding and automatic feedback generation for intelligent tutoring in basic electricity and electronics. *International Journal of Artificial Intelligence in Education, 24*(3), 284–332.

Easterday, M. W., Aleven, V., Scheines, R., & Carver, S. M. (2009). Constructing causal diagrams to learn deliberation. *International Journal of Artificial Intelligence in Education, 19*(4), 425–445.

Felton, M., & Kuhn, D. (2001). The development of argumentive discourse skills. *Discourse Processes, 32*(2&3), 135–153.

Fischer, F., Kollar, I., Stegmann, K., & Wecker, C. (2013). Toward a script theory of guidance in computer-supported collaborative learning. *Educational Psychologist, 48*(1), 56–66.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin, 76*(5), 378–382.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*(1), 119–139.

Freund, Y., & Schapire, R. E. (1999). A short introduction to Boosting. *Journal of Japanese Society for Artificial Intelligence, 14*(5), 771–780.

Forgy, C. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence, 19*(1), 17–37.

Friedman-Hill, E. (2003). *Jess in action: Java rule-based systems* (p. 38). Greenwich CT: Manning Publications.

Gigerenzer, G., Todd, P. M., & the ABC Research Group. (1999). *Simple heuristics that make us smart*. New York: Oxford University Press.

Godden D. M., & Walton, D. (2007). Advances in the theory of argumentation schemes and critical questions. *Informal Logic, 27*(3), 267–292.

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM, 35*(12), 61–70.

Goodman, B., Soller, A., Linton, F., & Gaimari, R. (1998). Encouraging student reflection and articulation using a learning companion. *International Journal of Artificial Intelligence in Education, 9*(3–4), 237–255.

Goodman, B. A., Linton, F. N., Gaimari, R. D., Hitzeman, J. M., Ross, H. J., & Zarrella, G. (2005). Using dialogue features to predict trouble during collaborative learning. *User Modeling and User-Adapted Interaction*, *16*(1–2), 85–134.

Gordon, T. F. (1996). Computational dialectics. In P. Hoschka (Ed.), *Computers as assistants – A new generation of support systems* (pp. 186–203). Mahwah, N.J: Lawrence Erlbaum Associates

Gordon, T. F., Prakken, H., & Walton, D. (2007). The Carneades model of argument and burden of proof. *Artificial Intelligence, 171*(10–15), 875–896.

Graesser, A. C., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N., & the Tutoring Research Group. (2000). Using latent semantic analysis to evaluate the contributions of students in AutoTutor. *Interactive Learning Environments, 8*(2), 129–147.

Graesser, A. C., Person, N. K., Harter, D., & the Tutoring Research Group. (2001). Teaching tactics and dialog in AutoTutor. *International Journal of Artificial Intelligence and Education, 12*(3), 257–279.

Grice, H. P. (1975). Logic and conversation. In P. Cole and J. J. Morgan (Eds.), *Syntax and semantics, Volume 3: Speech acts* (pp. 41– 58). New York: Academic Press.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: An update. *SIGKDD Explorations, 11*(1), 10–18.

Harrell, M. (2008). No computer program required: Even pencil-and-paper argument mapping improves critical-thinking skills. *Teaching Philosophy, 31*(4), 351–374.

Harrer, A., Hever, R., & Ziebarth, S. (2007). Empowering researchers to detect interaction patterns in e-collaboration. In R. Luckin, K. Koedinger, & J. Greer (Eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (pp. 503–510), Amsterdam, the Netherlands: IOS Press.

Harrer, A., Malzahn, N., & Wichmann, A. (2008). The remote control approach – An architecture for adaptive scripting across collaborative learning environments. *Journal of Universal Computer Science, 14*(1), 148–173.

Hattie, J., & Gan, M. (2011). Instruction based on Feedback. In R. E. Mayer (Ed.), *Handbook of research on learning and instruction* (pp. 249–271), New York: Routledge.

Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning, 11*(1), 63–90.

Hoppe, H. U. (1995). The use of multiple student modeling to parameterize group learning. In J. Greer (Ed.), *Proceedings of the 7th World Conference on Artificial Intelligence in Education* (pp. 234–241). Charlottesville, VA: AACE.

Hurley, P. J. (2011). *A concise introduction to logic (11th edition)*. Boston, MA: Wadsworth.

Israel, J., & Aiken, R. (2007). Supporting collaborative learning with an intelligent web-based system. *International Journal of Artificial Intelligence in Education, 17*, 3–40.

Janssen, J., Erkens, G., Kirschner, P., & Kanselaar, G. (2010). Effects of representational guidance during computer-supported collaborative learning. *Instructional Science, 38*(1), 59–88.

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem – A systematic study. *Intelligent Data Analysis, 6*(5), 429–450.

Jermann, P., & Dillenbourg, P. (2003). Elaborating new arguments through a CSCL script. In J. Andriessen, M. Baker, & D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments* (pp. 205–226). Dordrecht, the Netherlands: Kluwer.

Jermann, P. (2004). *Computer support for interaction regulation in collaborative problem-solving*. Doctoral Dissertation, University of Geneva.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In C. Nedellec, & C. Rouveirol (Eds.) *Proceedings of the 10th European Conference on Machine Learning* (pp. 137–142). Berlin, Germany: Springer.

Johnson, D. W., Maruyama, G., Johnson, R., Nelson, D., & Skon, L. (1981). Effects of cooperative, competitive, and individualistic goal structures on achievement: A meta-analysis. *Psychological Bulletin, 89*(1), 47–62.

Johnson, D. W., & Johnson, R. T. (1991). *Learning together and alone. Englewood Cliffs*, NJ: Prentice Hall.

Johnson, D. W., & Johnson, R. T. (1994). Constructive conflict in the schools. *Journal of Social Issues, 50*(1), 117–137.

Johnson, D. W., & Johnson, R. T. (2009). An educational psychology success story: Social interdependence theory and cooperative learning. *Educational Researcher*, *38*(5), 365–379.

Kahneman, D. (2011). *Thinking, fast and slow*. London, U.K.: Penguin Books.

Kim, J., Shaw, E., Ravi, S., Tavano, E., Arromratana, A., & Sarda, P. (2008). Scaffolding on-line discussions with past discussions: An analysis and pilot study of pedabot. In B. P. Woolf, E. Aimeur, R. Nkambou & S. Lajoie (Eds.), *Proceedings of the 9th International Conference on Intelligent Tutoring Systems* (pp. 343–352), Berlin, Germany: Springer.

Kintsch, W. (1994). Text comprehension, memory, and learning. *American Psychologist, 49*(4), 294–303.

Kirschner, P. A., Buckingham Shum, S. J., & Carr, C. S. (Eds.). (2003). *Visualizing argumentation: software tools for collaborative and educational sense-making*. London: Springer.

Kohavi, R., & John, G. H. (1998). The wrapper approach. In H. Liu, & H. Motoda (Eds.), *Feature extraction, construction and selection* (pp. 33–50). Dordrecht, the Netherlands: Kluwer.

Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P., & Fischer, F. (2007). Specifying computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning, 2*(2–3), 211–224.

Kollar, I., Fischer, F., & Hesse, F. W. (2006). Collaboration scripts: A conceptual analysis. *Educational Psychology Review, 18*(2), 159–185.

Koschmann, T. (1996). Paradigm Shifts and Instructional Technology. In T. Koschmann (Ed.), *CSCL: Theory and practice of an emerging paradigm* (pp. 1–23). Mahwah, NJ: Lawrence Erlbaum.

Krippendorff, K. (1980). *Content analysis: An introduction to its methodology*. Thousand Oaks, CA: Sage Publications.

Kuhn, D., Amsel, E., & O’Loughlin, M. (1988). *The development of scientific thinking skills*. Orlando, FL: Academic Press.

Kuhn, D. (1991). *The skills of argument*. New York: Cambridge University Press.

Kuhn, D., Shaw, V., & Felton, M. (1997). Effects of dyadic interaction on argumentive reasoning. *Cognition and Instruction, 15*(3), 287–315.

Kuhn, D., Cheney, R., & Weinstock, M. (2000). The development of epistemic understanding. *Cognitive Development, 15*(3), 309–328.

Kuhn, D., & Udell, W. (2003). The development of argument skills. *Child Development, 74*(5), 1245–1260.

Kuhn, D. (2005). *Education for thinking*. Cambridge, MA: Harvard University Press.

Kumar, R., Rosé, C.P., Wang, Y.-C., Joshi, M., & Robinson, A. (2007). Tutorial dialogue as adaptive collaborative learning support. In R. Luckin, K. R. Koedinger & J. Greer (Eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (pp. 383–390). Amsterdam, the Netherlands: IOS Press.

Kumar, R., Gweon, G., Joshi, M., Cui, Y., & Rosé, C. P. (2007b). Supporting students working together on math with social dialogue. Paper presented at the *SLaTE Workshop on Speech and Language Technology in Education*.

Kumar, R., Ai, H., Beuth, J. L., & Rosé, C. P. (2010). Socially capable conversational tutors can be effective in collaborative learning situations. In Aleven, V., Kay, J., & Mostow, J. (Eds.), *Proceedings of the International Conference on Intelligent Tutoring Systems, LNCS 6094* (pp. 156–164). Berlin: Springer.

Kumar, R., Beuth, J., & Rosé, C. P. (2011). Conversational strategies that support idea generation productivity in groups. In H. Spada, G. Stahl, N. Miyake, & N. Law (Eds.), *Connecting computer-supported collaborative learning to policy and practice: CSCL 2011 Conference Proceedings. Volume I – Long papers* (pp. 398–405). International Society of the Learning Sciences.

Kumar, R., & Rosé, C. P. (2011). Architecture for building conversational agents that support collaborative learning, *IEEE Transactions on Learning 4*(1), 21–34.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes, 25*(2-3), 259–284.

Landis J. R., & Koch G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*, 159–174.

Lazonder, A. W., Wilhelm, P., & Ootes, S. A. W. (2003). Using sentence openers to foster student interaction in computer-mediated learning environments. *Computers & Education, 41(3)*, 291–308.

Levinson, S. C. (1983). *Pragmatics*. Cambridge, U. K.: Cambridge University Press.

Linn, M. C., Clark, D., & Slotta, J. D. (2003). Wise design for knowledge integration. *Science Education*, *87*(4), 517–538.

Loll, F. (2012). *Domain-independent support for computer-based education of argumentation skills*. Doctoral Dissertation. Clausthal University of Technology, Germany, Department of Informatics.

Loll, F., Pinkwart, N., Scheuer, O., & McLaren, B. M. (2012). How tough should it be? Simplifying the development of argumentation systems using a configurable platform. In N. Pinkwart, & B. M. McLaren (Eds.), *Educational technologies for teaching argumentation skills* (pp. 169–197). Bentham Science Publishers.

Loll, F., & Pinkwart, N. (2013). LASAD: Flexible representations for computer-based collaborative argumentation. *International Journal of Human-Computer Studies, 71*(1), 91–109.

Lomborg, B. (2007). Conclusion: Making our top priorities cool. In: B. Lomborg, *Cool it: The skeptical environmentalist's guide to global warming*. Alfred A. Knopf, New York.

Loui, R. P., Norman, J., Altepeter, J., Pinkard, D., Linsday, J., & Foltz, M. (1997). Progress on room 5: A testbed for public interactive semi-formal legal argumentation. In *Proceedings of the 6th International Conference on Artificial Intelligence and Law* (pp. 207–214), New York: ACM Press.

Lund, K., Molinari, G., Séjorné, A., & Baker, M. (2007). How do argumentation diagrams compare when student pairs use them as a means for debate or as a tool for representing debate? *International Journal of Computer-Supported Collaborative Learning, 2*(2–3), 273–295.

Lynch, C. Ashley, K. D. Pinkwart, N., & Aleven, V. (2009). Concepts, structures, and goals: Redefining ill-definedness. *International Journal of Artificial Intelligence in Education, 19*(3). 253–266.

Lynch, C. & Ashley, K. D. (2012). Modeling student arguments in research reports. In V. G. Duffy (Ed.), *Advances in human factors and ergonomics 2012: Proceedings of the 4th AHFE Conference* (pp. 191–201), CRC Press.

Magnisalis, I., Demetriadis, S., & Karakostas, A. (2011). Adaptive and intelligent systems for collaborative learning support: A review of the field. *IEEE Transactions on Learning Technologies, 4*(1), 5–20.

Marcus, S. L., & Rips, L. J. (1979). Conditional reasoning. *Journal of Verbal Learning & Verbal Behavior, 18*(2), 199–233.

Mason, L., & Scirica, F. (2006). Prediction of students' argumentation skills about controversial topics by epistemological understanding. *Learning & Instruction, 16*(5), 492–509.

Mavrikis, M., Dragon, T., Abdu, R., Harrer, A., De Groot, R., & McLaren, B. (2012). Learning to learn together through planning, discussion and reflection on microworld-based challenges. In A. Ravenscroft, S. Lindstaedt, C. Delgado Kloos, & D. Hernández-Leo. (Eds.), *21st Century Learning for 21st Century Skills, Proceedings of the 7th European Conference on Technology Enhanced Learning, LNCS 7563* (pp. 483–488). Berlin: Springer.

Mayfield, E., Adamson, D., & Rosé, C. P. (2012). Hierarchical conversation structure prediction in multi-party chat. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue* (pp. 60–69). Association for Computational Linguistics.

McAlister, S. (2004) *Dialectics and design for online peer discussion*. Doctoral Dissertation. Institute of Educational Technology, Open University, U.K.

McAlister, S., Ravenscroft, A., & Scanlon, E. (2004). Combining interaction and context design to support collaborative argumentation using a tool for synchronous CMC. *Journal of Computer Assisted Learning: Special Issue: Developing Dialogue for Learning, 20*(3), 194–204.

McCallum, A., & Nigam K. (1998). A comparison of event models for Naïve Bayes text classification. In M. Sahami, M. Craven, T. Joachims, & A. McCallum (Eds.), *AAAI-98 Workshop on Learning for Text Categorization*, (pp. 41–48). Menlo Park, CA: AAAI Press.

McLaren, B. M., Scheuer, O., De Laat, M., Hever, R., De Groot, R., & Rosé, C. (2007). Using machine learning techniques to analyze and support mediation of student e-discussions. In R. Luckin,

K. R. Koedinger, & J. Greer (Eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (pp. 331–340), IOS Press, Amsterdam.

McLaren, B. M., Scheuer, O., & Mikšátko, J. (2010). Supporting collaborative learning and e-discussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education, 20*(1), 1–46.

McManus, M., & Aiken, R. (1995). Monitoring computer-based problem solving. *Journal of Artificial Intelligence in Education, 6*(4), 307–336.

Means, M. L., & Voss, J. F. (1996). Who reasons well? Two studies of informal reasoning among children of different grade, ability, and knowledge levels. *Cognition and Instruction, 14*(2), 139–178

Mercer, N. (1996). The quality of talk in children's collaborative activity in the classroom. *Learning and Instruction, 6*(4), 359–377.

Mercer, N. (2013). The social brain, language, and goal-directed collective thinking: A social conception of cognition and its implications for understanding how we think, teach, and learn. *Educational Psychologist, 48*(3), 148–168.

Merritt, D. (1989). *Building expert systems in Prolog*. Berlin: Springer.

Michaels, S., O'Connor, C., Hall, M., & Resnick, L. (2002). *Accountable Talk: Classroom conversation that works* (CD-ROM set). Pittsburgh: University of Pittsburgh.

Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., & Euler, T. (2006).YALE: Rapid prototyping for complex data mining tasks. In L. Ungar, M. Craven, D. Gunopulos, & T. Eliassi-Ras (Eds.), *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 935–940), New York: ACM.

Mikšátko, J. (2007). *Using machine learning techniques to analyze and recognize complex patterns of student e-discussions*. Master Thesis. Charles University, Prague.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM, 38*(11), 39–41.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw Hill.

Morrison, D., & Collins, A. (1996). Epistemic fluency and constructivist learning environments. In B. G. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design* (pp. 107–119), Englewood Cliffs NJ: Educational Technology Publications.

Mu, J., Stegmann, K., Mayfield, E., Rosé, C., & Fischer, F. (2012). The ACODEA framework: Developing segmentation and classification schemes for fully automatic analysis of online discussions. *International Journal of Computer-Supported Collaborative Learning, 7*(2), 285–305.

Mugny, G., & Doise, W. (1978). Socio-cognitive conflict and structure of individual and collective performances. *European Journal of Social Psychology, 8*(2), 181–192.

Menekse, M., Stump, G. S., Krause, S., & Chi, M. T. H. (2013). Differentiated overt learning activities for effective instruction in engineering classrooms. *Journal of Engineering Education, 102*(3), 346–374.

Munneke, L., van Amelsvoort, M., & Andriessen, J. (2003). The role of diagrams in collaborative argumentation-based learning. *International Journal of Educational Research, 39*(1–2), 113–131.

Mühlpfordt, M., & Wessner, M. (2005). Explicit referencing in chat supports collaborative learning. In T. Koschmann, D. Suthers, & T. W. Chan (Eds.), *Proceedings of the Conference on Computer Supported Collaborative Learning* (pp. 460–469), Mahwah, NJ: Lawrence Erlbaum.

Murray, T., Blessing, S., & Ainsworth, S. (Eds.). (2003). *Authoring tools for advanced technology learning environments*. Dordrecht: Kluwer Academic Publishers.

Murray, T., Woolf, B., & Marshall, D. (2004). Lessons learned from authoring for inquiry learning: A tale of authoring tool evolution. In J. C. Lester, R. M. Vicari, & F. Paraguaçu (Eds.), *Proceedings of the 7th International Conference on Intelligent Tutoring Systems* (pp. 197–206). Berlin: Springer.

Murray, R. C., VanLehn, K., & Mostow, J. (2004b). Looking ahead to select tutorial actions: A decision-theoretic approach. *International Journal of Artificial Intelligence in Education, 14*(3), 235–278.

Muller Mirza, N., Tartas, V., Perret-Clermont, A.-N., & De Pietro, J.-F. (2007). Using graphical tools in a phased activity for enhancing dialogical skills: An example with digalo. *International Journal of Computer-Supported Collaborative Learning, 2*(2–3), 247–272

Ng, H. T., Goh, W. B., & Low, K. L. (1997). Feature selection, perceptron learning, and a usability case study for text categorization. In N. J. Belkin, A. D. Narasimhalu, & P. Willett (Eds.), *Proceedings of SIGIR-97, 20th ACM International Conference on Research and Development in Information Retrieval* (pp. 67–73). New York: ACM Press.

Norman, D. (1988). *The psychology of everyday things*. New York: Basic Book.

Noss, R., Poulovassilis, A., Geraniou, E., Gutierrez-Santos, S., Hoyles, C., Kahn, K., Magoulas, G. D., & Mavrikis, M. (2012). The design of a system to support exploratory learning of algebraic generalization. *Computers & Education, 59*(1), 63–81.

Novak, J. D. (1990). Concept maps and vee diagrams: Two metacognitive tools for science and mathematics education. *Instructional Science, 19*(1), 29–52.

Nussbaum, E. M., Hartley K., Sinatra, G. M., Reynolds, R. E., & Bendixen, L. D. (2002). Enhancing the quality of on-line discussions. Paper presented at the *Annual Meeting of the American Educational Research Association*.

Nussbaum, E. M., Winsor, D. L., Aqui, Y. M., & Poliquin, A. M. (2007). Putting the pieces together: Online argumentation vee diagrams enhance thinking during discussions. *International Journal on Computer-Supported Collaborative Learning, 2*(4), 479–500.

Nussbaum, E. M. (2008). Collaborative discourse, argumentation, and learning: Preface and literature review. *Contemporary Educational Psychology*, *33*(3), 345–359.

O'Donnell, A. M., & Dansereau, D. F. (1992). Scripted cooperation in student dyads: A method for analyzing and enhancing academic learning and performance. In R. Hertz-Lazarowitz, & N. Miller (Eds.), *Interaction in cooperative groups: The theoretical anatomy of group learning* (pp. 120–141). London: Cambridge University Press

O'Donnell, A. M., Dansereau, D. F., & Hall, R. H. (2002). Knowledge maps as scaffolds for cognitive processing. *Educational Psychology Review*, *14*(1), 71–86.

Oh, S. & Jonassen, D. H. (2007). Scaffolding online argumentation during problem solving. *Journal of Computer Assisted Learning, 23*(2), 95–110.

Osborne, J., Erduran, S., & Simon, S. (2004). Enhancing the quality of argumentation in school science. *Journal of Research in Science Teaching*, *41*(10), 994–1020.

Osborne, J. (2010). Arguing to learn in science: The role of collaborative, critical discourse. *Science, 328*(5977), 463–466,

Pea, R. D. (2004). The social and technological dimensions of scaffolding and related theoretical concepts for learning, education, and human activity. *The Journal of the Learning Sciences, 13*(3), 423–451.

Perkins, D., Allen, R., & Hafner, J. (1983). Difficulties in everyday reasoning. In W. Maxwell (Ed.), *Thinking: The expending frontier* (pp. 177–189). Philadelphia, PA: Franklin Institute Press.

Perkins, D. N. (1985). Postprimary education has little impact on informal reasoning. *Journal of Educational Psychology, 77*(5), 562–571.

Perkins, D. N., & Salomon, G. (1989). Are cognitive skills context-bound? *Educational Researcher, 18*(1), 16–25.

Perkins, D. N., Jay, E., & Tishman, S. (1993). Beyond abilities: A dispositional theory of thinking. *Merrill-Palmer Quarterly, 39*(1), 1–21.

Perkins, D., Tishman, S., Ritchhart, R., Donis, K., & Andrade, A. (2000). Intelligence in the wild: A Dispositional view of intellectual traits. *Educational Psychology Review, 12*(3), 269–293.

Pfister, H. R., & Mühlpfordt, M. (2002). Supporting discourse in a synchronous learning environment: The learning protocol approach. In G. Stahl (Ed.), *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL community* (pp. 581–582). International Society of the Learning Sciences.

Piaget, J. (1972). Intellectual evolution from adolescence to adulthood. *Human Development, 15*(1), 1–12.

Piaget, J. (1985). *The equilibrium of cognitive structures: The central problem of intellectual development*. Chicago: University of Chicago Press.

Pinkwart, N., Aleven, V., Ashley, K., & Lynch, C. (2006). Schwachstellenermittlung und Rückmeldungsprinzipien in einem intelligenten Tutorensystem für juristische Argumentation. In M.

Mühlhäuser, G. Rößling, & R. Steinmetz (Eds.), *GI Lecture Notes in Informatics - Tagungsband der 4. e-Learning Fachtagung Informatik*.

Pinkwart, N., Ashley, K., Lynch, C., & Aleven, V. (2008). Graph grammars: An ITS technology for diagram representations. In D. Wilson, & H. C. Lane (Eds.), *Proceedings of 21st International. FLAIRS Conference.* (pp. 433–438).

Pinkwart, N., Ashley, K. D., Lynch, C., & Aleven, V. (2009). Evaluating an intelligent tutoring dystem for making legal arguments with hypotheticals. *International Journal of Artificial Intelligence in Education, 19*(4), 401–424.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Mateo, CA: Morgan Kaufmann Publishers.

Ranney, M., & Schank, P. (1998). Toward an integration of the social and the scientific: Observing, modeling, and promoting the explanatory coherence of reasoning. In S. Read, & L. Miller (Eds.), *Connectionist models of social reasoning and social behavior* (pp. 245–274). Mahwah, NJ: Lawrence Erlbaum.

Ravenscroft, A. (2007). Promoting thinking and conceptual change with digital dialogue games. *Journal of Computer Assisted Learning, 23*, 453–465.

Reed, C., Walton, D., & Macagno, F. (2007). Argument diagramming in logic, law, and artificial intelligence. *The Knowledge Engineering Review 22*(1), 87–109.

Resnick, L. B., Michaels, S., & O'Connor, C. (2010). How (well structured) talk builds the mind. In D. Preiss, & R. Sternberg (Eds.), *Innovations in educational psychology: Perspectives on learning, teaching and human development* (pp. 163–194). New York: Springer.

Rewey, K. L., Dansereau, D. F., Skaggs, L. P., & Hall, R. H. (1989). Effects of scripted cooperation and knowledge maps on the processing of technical material. *Journal of Educational Psychology, 81*(4), 604–609.

Rewey, K. L., Dansereau, D. F., Dees, S. M., Skaggs, L. P., & Pitre, U. (1992). Scripted cooperation and knowledge map supplements: Effects on the recall of biological and statistical information. *Journal of Experimental Education, 60*(2), 93–107.

Reznitskaya, A., Kuo, L.-J., Clark, A.-M., Miller, B., Jadallah, M., Anderson, R., & Nguyen-Jahiel, K. (2009). Collaborative reasoning: a dialogic approach to group discussions. *Cambridge Journal of Education, 39*(1), 29–48.

Ritchhart, R., & Perkins, D. N. (2005). Learning to think: The challenges of teaching thinking. In K. J. Holyoak, & R. G. Morrison (Eds.), *The Cambridge handbook of thinking and reasoning* (pp. 775–802). New York: Cambridge University Press.

Rittel, H. W. T., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Sciences, 4*, 155–169.

Robertson, J., Good, J., & Pain, H. (1998). BetterBlether: The design and evaluation of a discussion tool for education. *International Journal of Artificial Intelligence in Education 9*(2), 219–236.

Rogoff, B. (1990). *Apprenticeship in thinking: Cognitive development in social context*. New York: Oxford University Press.

Rogoff, B. (1995). Observing sociocultural activity on three planes: Participatory appropriation, guided participation, and apprenticeship. In J. V. Wertsch, P. del Rio, & A. Alvarez (Eds.), *Sociocultural studies of mind* (pp. 139–164). Cambridge, UK: Cambridge University Press.

Roscoe, R. D., & Chi, M. T. H. (2007). Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors' explanations and questions. *Review of Educational Research, 77*(4), 534–574.

Rosé, C. P., & Torrey, C. (2005). Interactivity versus expectation: Eliciting learning oriented behavior with tutorial dialogue systems. In *Proceedings of Interact* (pp. 323–336), Berlin: Springer.

Rosé, C., Wang, Y.-C., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., & Fischer, F. (2008). Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. *International Journal of Computer-Supported Collaborative Learning, 3*(3), 237–271.

Rummel, N., & Spada, H. (2005). Learning to collaborate: An instructional approach to promoting collaborative problem-solving in computer-mediated settings. *The Journal of the Learning Sciences, 14*(2), 201–241.

Sacks, H., Schegloff, E. A., & Jefferson, G. (1974). A simplest systematics for the organization of turn-taking in conversation. *Language, 50*(4), 696–735.

Sadler, T. D., & Zeidler, D. L. (2005). The significance of content knowledge for informal reasoning regarding socioscientific issues: Applying genetics knowledge to genetic engineering issues. *Science Education, 89*(1), 71–93.

Salmon, G. (2004). *E-moderating: The key to teaching and learning online* (2nd edition). London: Routledge Falmer.

Salomon, G. (Ed.). (1997). *Distributed cognitions: Psychological and educational considerations.* Cambridge University Press, Cambridge, U.K.

Sampson, V., & Blanchard, M. R. (2012). Science teachers and scientific argumentation: Trends in views and practice. *Journal of Research in Science Teaching, 49*(9), 1122–1148.

Scardamalia, M., & Bereiter, C. (1983). The development of evaluative, diagnostic, and remedial capabilities in children's composition. In M. Martlew (Ed.), *The psychology of written language: Developmental and educational perspectives* (pp. 67–95). London: Wiley.

Schank, P. K. (1995). *Computational tools for modeling and aiding reasoning: Assessing and applying the theory of explanatory coherence.* Doctoral Dissertation. University of California, Berkeley, USA.

Schapire, R. E., & Singer, Y. (2000). BoosTexter: A boosting-based system for text categorization. *Machine Learning, 39*(2–3), 135–168.

Schellens, T., van Keer, H., de Wever, B., & Valcke, M. (2007). Scripting by assigning roles: Does it improve knowledge construction in asynchronous discussion groups? *International Journal of Computer Supported Collaborative Learning, 2*(2–3), 225–246.

Schellens, T., & Fischer, F. (Eds.). (2013). Computer-supported collaborative learning (CSCL): Exploring synergetic scaffolding and scripting. Symposium at the *15th Biennial EARLI Conference for Research on Learning and Instruction*.

Scheuer, O., Loll, F., Pinkwart, N., & McLaren, B. M. (2010). Computer-supported argumentation: A review of the state-of-the-art. *International Journal of Computer-Supported Collaborative Learning, 5*(1), 43–102. doi:10.1007/s11412-009-9080-x

Scheuer, O, McLaren, B. M., Harrell, M., & Weinberger, A. (2011). Scripting collaboration: What affects does it have on atudent argumentation? In T. Hirashima et al. (Eds.), *Proceedings of the International Conference on Computers in Education* (pp. 181–188). Asia-Pacific Soc. for Computers in Education.

Scheuer, O., McLaren, B. M., & Weinberger, A. (2013). Automated and adaptive support for educational discussions: Results to guide in making this a reality. In N. Rummel, M. Kapur, M. Nathan, & S. Puntambekar (Eds.), *To see the world and a grain of sand: Learning across levels of space, time, and scale: CSCL 2013 Conference Proceedings Volume 2 – Short papers, panels, posters, demos & community events* (pp. 349–350). International Society of the Learning Sciences.

Scheuer, O., & McLaren, B. M. (2013). CASE: A configurable argumentation support engine. *IEEE Transactions on Learning Technologies, 6*(2), 144–157.

Scheuer, O., McLaren, B. M., Weinberger, A., & Niebuhr, S. (2014). Promoting critical, elaborative discussions through a collaboration script and argument diagrams. *Instructional Science, 42*(2), 127–157. doi:10.1007/s11251-013-9274-5

Schegloff, E. A., & Sacks, H. (1973). Opening up closings. *Semiotica, 8*(4), 289–327.

Schler, J., Koppel, M., Argamon, S., & Pennebaker, J. (2006). Effects of age and gender on blogging. Paper presented at the *AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs* (pp. 199–205), Stanford, CA.

Schneider, J., Groza, T., & Passant, A. (2013). A review of argumentation for the social semantic web. *Semantic Web, 4*(2), 159–218.

Schwarz, B. B., Neuman, Y., & Biezuner, S. (2000). Two Wrongs May Make a Right ... If They Argue Together! *Cognition and Instruction*, *18*(4), 461–494.

Schwarz, B. B., & Glassner, A. (2007). The role of floor control and of ontology in argumentative activities with discussion-based tools. *International Journal of Computer-Supported Collaborative Learning*, *2*(4), 449–478.

Searle, J. R. (1969). *Speech acts*. Cambridge: Cambridge University Press.

Searle, J. R. (1976). A classification of illocutionary acts. *Language in Society, 5*(1), 1–23.

Self, J. (1990). Bypassing the intractable problem of student modelling. In C. Frasson, & G. Gauthier (Eds.), *Intelligent tutoring systems: At the crossroad of artificial intelligence and education* (107–123), Norwood, NJ: Ablex.

Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research, 78*(1), 153–189.

Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics, 69*(1), 99–118.

Simon, S., Erduran, S., & Osborne, J. (2006). Learning to teach argumentation: Research and development in science classrooms. *International Journal of Science Education, 28*(2–3), 235–260.

Slavin, R. E. (1996). Research on cooperative learning and achievement: What we know, what we need to know. *Contemporary Educational Psychology, 21*(1), 43–69.

Soller, A. (2001). Supporting social interaction in an intelligent collaborative learning system. *International Journal of Artificial Intelligence in Education, 12*, 40–62.

Soller, A. (2004). Computational modeling and analysis of knowledge sharing in collaborative distance learning. *User Modeling and User-Adapted Interaction, 14*(4), 351–381.

Soller, A., Monés, A. M., Jermann, P., & Mühlenbrock, M. (2005). From mirroring to guiding: A review of the state of the art technology for supporting collaborative learning. *International Journal of Artificial Intelligence in Education, 15*, 261–290.

Stahl, G. (2006). *Group cognition: Computer support for building collaborative knowledge.* Cambridge, MA: MIT Press.

Stahl, G., Koschmann, T., & Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. In R. K. Sawyer (Ed.), *Cambridge handbook of the learning sciences* (pp. 409-426). Cambridge, UK: Cambridge University Press.

Stahl, G. (2007). CSCL and its flash themes. *International Journal of Computer-Supported Collaborative Learning, 2*(4), 359–362.

Stark, R., & Krause, U. M. (2006). Konzeption einer computerbasierten Lernumgebung zur Förderung von Kompetenzen zum wissenschaftlichen Argumentieren. In G. Krampen, H. Zayer (Eds.), *Didaktik und Evaluation in der Psychologie* (pp. 218–230). Göttingen: Hogrefe.

Stegmann, K., Weinberger, A., & Fischer, F. (2007). Facilitating argumentative knowledge construction with computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning, 2*(4), 421–447.

Stegmann, K., Wecker, C., Weinberger, A., & Fischer, F. (2012). Collaborative argumentation and cognitive elaboration in a computer-supported collaborative learning environment. *Instructional Science, 40*(2), 297–323.

Stein, N. L., & Miller, C. A. (1993). A theory of argumentative understanding: Relationships among position preference, judgments of goodness, memory, and reasoning. *Argumentation, 7*(2), 183–204.

Stock, W. G. (1982). E. M. Barth, E. C. W. Krabbe: From axiom to dialogue (critique). *International Classification, 10*(3), 166–169.

Strijbos, J. W., Martens, R. L., Jochems, W. M. G., & Broers, N. J. (2004). The effect of functional roles on group efficiency: Using multilevel modeling and content analysis to investigate computer-supported collaboration in small groups. *Small Group Research, 35*(2), 195–229.

Suthers, D. D., Toth, E., & Weiner, A. (1997). An integrated approach to implementing collaborative inquiry in the classroom. In R. Hall, N. Miyake, & N. Enyedy (Eds.), *Proceedings of the 2nd International Conference on Computer Support for Collaborative Learning* (pp. 272–279). Toronto, Ontario: University of Toronto Press.

Suthers, D. D., Connelly, J., Lesgold, A., Paolucci, M., Toth, E. E., Toth, J., & Weiner, A. (2001). Representational and advisory guidance for students learning scientific inquiry. In K. D. Forbus, & P. J. Feltovich (Eds.), *Smart machines in education: The coming revolution in educational technology* (pp. 7–35). Menlo Park, CA: AAAI/MIT Press.

Suthers, D. D. (2003). Representational guidance for collaborative inquiry. In J. Andriessen, M. J. Baker, & D. D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments* (pp. 27–46). Dordrecht, the Netherlands: Kluwer Academic Publishers.

Suthers, D. D., & Hundhausen, C. (2003). An experimental study of the effects of representational guidance on collaborative learning processes. *The Journal of the Learning Science*s, *12* (2), 183–219.

Suthers, D. D., Hundhausen, C. D., & Girardeau, L. E. (2003). Comparing the roles of representations in face-to-face and online computer supported collaborative learning. *Computers & Education, 41*(4), 335–351.

Suthers, D. D. (2006). Technology affordances for intersubjective meaning making: A research agenda for CSCL. *International Journal of Computer-Supported Collaborative Learning, 1*(3), 315–337.

Suthers, D. D. (2008). Empirical studies of the value of conceptually explicit notations in collaborative learning. In A. Okada, S. Buckingham Shum, & T. Sherborne (Eds.), *Knowledge cartography* (pp. 1–23), London: Springer.

Suthers, D. D., Vatrapu, R., Medina, R., Joseph, S., & Dwyer, N. (2008). Beyond threaded discussion: Representational guidance in asynchronous collaborative learning environments. *Computers & Education, 50*(4), 1103–1127.

Tabak, I. (2004). Synergy: A complement to emerging patterns of distributed scaffolding. *The Journal of the Learning Sciences, 13*(3), 305–335.

Tchounikine, P. (2008). Operationalizing macro-scripts in CSCL technological settings. *International Journal of Computer-Supported Collaborative Learning, 3*(2), 193–233.

Tishman, S., Jay, E., & Perkins, D. N. (1993). Teaching thinking dispositions: From transmission to enculturation. *Theory into Practice, 32*(3), 147–153.

Toulmin, S. E. (1958). *The uses of argument.* Cambridge, UK: Cambridge University Press.

Toulmin, S. E. (2003). *The uses of argument (updated edition).* Cambridge, UK: Cambridge University Press.

Tsovaltzi, D., Rummel, N., McLaren, B. M., Pinkwart, N., Scheuer, O., Harrer, A., et al. (2010). Extending a virtual chemistry laboratory with a collaboration script to promote conceptual learning. *International Journal of Technology Enhanced Learning, 2*(1–2), 91–110.

Twardy, C. R. (2004). Argument maps improve critical thinking. *Teaching Philosophy, 27*(2), 95–116.

Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science, 185*(4157), 1124–1131.

Valero Haro, A. (2013). *Using a graphical user interface to simplify the development of tutorial agents that support argumentation learning activities.* Master Thesis. Saarland University, Department of Computer Science.

Van Drie, J., Van Boxtel, C., Jaspers, J., & Kanselaar, G. (2005). Effects of representational guidance on domain specific reasoning in CSCL. *Computers in Human Behavior, 21*(4), 575–602.

Van Eemeren, F. H., Grootendorst, R., Snoeck Henkemans, F., Blair, J. A., Johnson, R. H., Krabbe, E. C. W., Plantin, C., Walton, D. N., Willard, C. A., Woods, J., & Zarefsky, D. (1996). *Fundamentals of argumentation theory.* Mahwah, NJ: Lawrence Erlbaum Associates.

Van Eemeren, F. H., & Grootendorst, R. (2004). *A systematic theory of argumentation: The pragma-dialectical approach*. Cambridge, UK: Cambridge University Press.

Van Gelder, T. (2002). Argument mapping with Reason!Able. *The American Philosophical Association Newsletter on Philosophy and Computers, 2* (1), 85–90.

Van Gelder, T. (2005). Teaching critical thinking: some lessons from cognitive science. *College Teaching, 53*, 41–46.

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education, 16*(3), 227–265.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist, 46*(4), 197–221.

Von Aufschnaiter, C., Erduran, S., Osborne, J., & Simon, S. (2008). Arguing to learn and learning to argue: Case studies of how students' argumentation relates to their scientific knowledge. *Journal of Research in Science Teaching, 45*(1), 101–131.

Voss, J. F., & Means, M. L. (1991). Learning to reason via instruction in argumentation. *Learning and Instruction, 1*(4), 337–350.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.

Walker, E., Rummel, N., & Koedinger, K. R. (2011). Designing automated adaptive support to improve student helping behaviors in a peer tutoring activity. *International Journal of Computer Supported Collaborative Learning, 6*(2), 279–306.

Walker, E., Rummel, N., & Koedinger, K. R. (2011b). Adaptive support for CSCL: Is it feedback relevance or increased accountability that matters? In H. Spada, G. Stahl, N. Miyake, & N. Law (Eds.), *Connecting computer-supported collaborative learning to policy and practice: CSCL 2011 Conference Proceedings. Volume I – Long papers* (pp. 334–342). International Society of the Learning Sciences.

Walker, E., Rummel, N., Walker, S., & Koedinger, K. (2012). Noticing relevant feedback improves learning in an intelligent tutoring system for peer tutoring. In S. A. Cerri, W. J. Clancey, G. Papadourakis, & K. Panourgia (Eds.), *Proceedings of the International Conference on Intelligent Tutoring System, LNCS 7315* (pp. 222–232). Berlin, Germany: Springer.

Walton, D. (1996). *Argumentation schemes for presumptive reasoning*. Mahwah, NJ: Lawrence Erlbaum.

Walton, D. (1999). *Appeal to popular opinion*. University Park, PA: Penn State Press.

Walton, D. (2005). Begging the question in arguments based on testimony. *Argumentation, 19*(1), 85–113.

Walton, D. (2008). *Informal logic: A pragmatic approach (2nd edition)*. Cambridge, U.K.: Cambridge University Press.

Walton, D., Reed, C., & Macagno, F. (2008). *Argumentation schemes*. Cambridge, U.K.: Cambridge University Press.

Webb, N. M. (1989). Peer interaction and learning in small groups. *International Journal of Educational Research, 13*(1), 21–39.

Wecker, C., & Fischer, F. (2011). From guided to self-regulated performance of domain-general skills: The role of peer monitoring during the fading of instructional scripts. *Learning and Instruction, 21*(6), 746–756.

Wegerif, R., Mercer, N., & Dawes, L. (1999). From social interaction to individual reasoning: An empirical investigation of a possible socio-cultural model of cognitive development. *Learning and Instruction, 9*(5), 493–516.

Wegerif, R. (2006). A dialogic understanding of the relationship between CSCL and teaching thinking skills. *International Journal of Computer Supported Collaborative Learning, 1*(1), 143–157.

Wegerif, R., McLaren, B. M., Chamrada, M., Scheuer, O., Mansour, N., Mikšátko, J., & Williams, M. (2010). Exploring creative thinking in graphically mediated synchronous dialogues. *Computers & Education, 54*(3), 613–621.

Wegerif, R. (2011). Towards a dialogic theory of how children learn to think. *Thinking Skills and Creativity, 6*(3), 179–190, Weinberger, A., Ertl, B., Fischer, F., & Mandl, H. (2005). Epistemic and social scripts in computer–supported collaborative learning. *Instructional Science*, *33*(1), 1–30.

Weinberger, A., & Fischer, F. (2006). A Framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers & Education, 46*(1), 71–95.

Weinberger, A., Stegmann, K., & Fischer, F. (2007). Knowledge convergence in collaborative learning: Concepts and assessment. *Learning and Instruction*, *17*(4)**,** 416–426.

Weinberger, A., Stegmann, K., & Fischer, F. (2010). Learning to argue online: Scripted groups surpass individuals (unscripted groups do not). *Computers in Human Behavior*, *26*(4), 506–515.

Weinberger, A. (2011). Principles of transactive computer-supported collaboration scripts. *Nordic Journal of Digital Literacy, 6*(3), 189–202.

Weiss, G. M. (2004). Mining with rarity: A unifying framework. *ACM SIGKDD Explorations Newsletter, 6*(1), 7–19.

Wigmore, J. H. (1931). *The principles of judicial proof* (2nd edition). Little, Brown & Co: Boston.

Witten, I. H., & Frank E. (2005). *Data mining: Practical machine learning tools and techniques* (2nd edition). San Francisco, CA: Morgan Kaufmann.

Woods, J., & Walton, D. (1977). Post hoc, ergo propter hoc. *The Reviews of Metaphysics, 30*(4), 569–593.

Woolf, B. P., Marshall, D., Mattingly, M., Lewis, J., Wright, S., Jellison, M., & Murray, T. (2003). Tracking student propositions in an inquiry system. In U. Hoppe, F. Verdejo, & J. Kay (Eds.), *Proceedings of the 11th International Conference on Artificial Intelligence in Education* (pp. 21–28), Amsterdam, the Netherlands: IOS Press.

Woolf, B. P. (2008). *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning.* San Francisco, CA: Elsevier Inc., Morgan Kaufmann.

Yuan, T., Moore, D., & Grierson, A. (2008). A human-computer dialogue system for educational debate: A computational dialectics approach. *International Journal of Artificial Intelligence in Education, 18*(1), 3–26.

Zloof, M. M. (1977). Query-by-example: A data base language. *IBM Systems Journal, 16*, 324–343.

Zohar, A., & Nemet, F. (2002). Fostering students' knowledge and argumentation skills through dilemmas in human genetics. *Journal of Research in Science Teaching, 39*(1), 35–62.