

scripted |

Volume 14, Issue 2, December 2017

Argument Invention with the Carneades Argumentation System

Douglas Walton and Thomas F. Gordon***



© 2017 Douglas Walton and Thomas F. Gordon
Licensed under a Creative Commons Attribution-NonCommercial-
NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

DOI: 10.2966/scrip.140217.168

Abstract

Argument invention (*inventio*) has traditionally been regarded as one of the five main components of rhetoric, but has remained an ambiguous, vague and highly contested concept, made even more confusing by its dependence on the Aristotelian topics, supposedly the places in which the rhetorical persuader can find arguments useful to support or attack a claim. The advent of two recently developed computational tools for argument invention, the Carneades Argumentation System and IBM's Watson Debater tool, calls for a rethinking of the notion of argument invention in line with the state of the art of formal and computational argumentation systems in artificial intelligence. The role of argumentation schemes is an important part of this investigation into argument invention.

Keywords

rhetoric, invention, legal argumentation, finding arguments, computational argument systems

* Centre for Research in Reasoning, Argumentation and Rhetoric, University of Windsor, Windsor, Ont., Canada, waltoncrrar@gmail.com

** Fraunhofer FOKUS, Berlin, Germany, thomas.gordon@fokus.fraunhofer.de

1 Introduction

Argument invention has been taken to be one of the five main parts of rhetoric since the times of Aristotle and Cicero, but now with the advent of computational tools to help a debater find pro and con arguments to support a designated thesis, new questions are raised about what argument invention is. Argument invention, or *inventio* as it is usually called in rhetoric, has been a philosophically contested concept for over two millennia, and has been acknowledged as ambiguous. In one sense it can refer to finding something that was already there, as in a discovery, but in another sense it can be restricted to finding something new, something that did not exist before. These problems have been compounded in the history of rhetoric by the connection between argument invention and the so-called topics, supposed to be the traditional tools for finding arguments according to such sources as Aristotle, Quintilian, Cicero and other notables.¹ The meaning of the term “topic” has had such a diversity of meanings over the centuries that the notion of argument invention itself remains difficult to untangle because of the heavy weight of historical baggage it carries.² The concept of argumentation schemes was inspired by the topics and was intended from the beginning to handle them. This paper takes us further by probing more deeply into the connection between schemes and argument invention. A legal example from Cicero’s classic work *De Inventione* is analysed to provide some evidence to support this approach.

Section 2 offers a quick overview of the historical beginnings of the subject of argument invention by Greek and Roman authors on rhetoric, showing how

¹ Fabrizio Macagno and Douglas Walton, “Argumentation Schemes and Topical Relations” in Giovanni Gobber and Andrea Rocci (eds.), *Language, Reason and Education* (Bern: Peter Lang, 2014) pp. 185-216.

² Fabrizio Macagno and Douglas Walton, “Classifying the Patterns of Natural Arguments” (2015) 48(1) *Philosophy and Rhetoric* 139-159.

it became an essentially contested concept during its successive interpretations. Because it was so closely tied in with the so-called “topics” representing places to find arguments, and because this notion changed so many times during the historical course of its development, argument invention was never defined clearly or precisely in a way that enables others to grasp or apply it as an effective tool for rhetoric. The notion of argument invention continues to this day to contain vagueness and ambiguities that make it difficult to approach without becoming encumbered and confused by its historical baggage. However, an example from Cicero is used to suggest that there may be some connection between the ancient topics and the present-day argumentation schemes.

Section 3 introduces the Watson Debater, a computational tool recently devised by IBM to help a human debater find arguments to support or attack a thesis. There is some doubt about whether Debater is an argument invention tool or merely an argument mining tool, and so Section 4 discusses what the purpose of argument mining is supposed to be and briefly outlines the main approaches to argument mining.

Currently it is a subject of some interest whether argument invention can be modelled using the formal argumentation systems currently being developed in AI and Law.³ Section 5 briefly surveys three of these models, showing that all of them can potentially assist a user in the task of argument construction, because they are all knowledge-based systems using an argument mapping tool. But only one of them, the Carneades Argumentation System, has implemented a tool specifically designed to assist a user with the task of argument invention. Section 6 outlines the Carneades procedure of argument invention whereby the user can invent arguments using a knowledge base and a set of argumentation schemes.

³ Kevin Ashley, *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age* (Cambridge: Cambridge University Press, 2017).

Sections 7 and 8 present three simple examples to illustrate how this procedure of argument invention is carried out. Section 9 gives a legal example. Section 10 offers some conclusions.

Debater extracts arguments from a natural language knowledge base using natural language processing methods and looks like a system of argument mining. The question explored in this paper is whether the two systems, Debater and Carneades, are trying to carry out the same goal (argument invention), or whether they trying to do something different. It is suggested that the prospects for building systems of argument invention could be enhanced by integrating them. The role of argumentation schemes in argument invention is discussed.

2 The ancient history of the topics in argument invention

Argument invention (*inventio*) is a technique going back to the ancient Greek Sophists, and from there to the later Greek philosophers, most notably Aristotle, and Roman rhetoricians and legal practitioners, most notably Cicero. It is well known that the ancients worked systematically on rhetorical methods of argument invention designed to help an arguer to support or attack a claim in a debate.⁴ Cicero, in *De Inventione*,⁵ claiming to follow Aristotle's view, divided the material of the art of rhetoric into five parts: invention, arrangement, expression, memory, and delivery. Invention is defined as "the discovery of valid or seemingly valid arguments to render one's cause plausible."⁶ Arrangement is the placement of the discovered arguments in the right order. Expression is the filling of the arguments that have been invented into the proper language.

⁴ George Kennedy, *The Art of Persuasion in Greece* (London: Routledge, 1963).

⁵ Marcus Cicero, *De Inventione* (Loeb Classical Library Edition, tr. Harry Hubbell, Cambridge, Mass: Harvard University Press, 1949), I, 6, 9-10.

⁶ *Ibid.*, I, 6.

Memory is the recall of words. Delivery is the use of voice and body to present the arguments in a suitable style.

However, throughout the long history of the study of rhetoric since ancient times, argument invention has proved to be an essentially contested concept that has been subject to successive interpretations through each historical tradition.⁷ It has been a source of frustration and difficulty for the modern argumentation theorist to try to make sense of these traditional doctrines in a precise or coherent enough way to make them practically useful.

To start with, the Latin word *inventio* is ambiguous.⁸ Discovery, such as the discovery of a new planet, finds something that was already there whereas invention, such as the invention of a new device described in a patent application, comes up with something new and different from what was here before. In English, the expression “finding an argument” as used in the literature on argumentation is also ambiguous in a comparable way. It can be used to describe the action of finding an argument in an existing text of discourse, such as a book, but it could also be used to include the action of using one’s imagination or one’s knowledge of an audience to come up with an argument that could be used to persuade them to do something or to accept a claim. The first meaning sounds very much like what we now call argument mining, whereas the second meaning sounds like something different.

Throughout the history of rhetoric, logic and philosophy, argument invention has been linked to Aristotle’s *Topics*.^{9,10} The meaning of the word

⁷ John Arthos, “Rhetorical Invention” (2017), Oxford Research Encyclopedia of Communication, available at <http://communication.oxfordre.com/view/10.1093/acrefore/9780190228613.001.0001/acrefore-9780190228613-e-42?print> (accessed 31 October 2017).

⁸ *Ibid.*, 1.

⁹ Aristotle, *Topics* (tr. E.S. Forster, Cambridge: Harvard University Press, 1939).

¹⁰ Macagno and Walton, “Argumentation Schemes and Topical Relations”, *supra* n. 1.

“topic” as a technical device of argument evaluation and invention in rhetoric, logic and philosophy has been hotly contested from ancient times through Quintilian, Cicero, Boethius, and on and on, up to Perelman and Olbrechts-Tyteca and beyond. The *tópos*, or what is generally taken to be its equivalent, the word *locus*, refers to some place or source where an argument can be found so that it can be retrieved for use in building an argument.¹¹ The term is closely related to argument invention, and indeed the *topoi* are supposedly the primary tools used in argument invention, but precisely how they are to be used for this purpose has never been established with any unanimity. Indeed, used as a technical term in rhetoric through the ages, the term “topic” has had “a bewildering diversity of meanings.”¹² The rhetoric scholar Michael Leff, who spent a distinguished career trying to rehabilitate the notion of topic as a coherent resource for the technique of argument invention that is so important for the field of rhetoric, gradually discovered that it proved to be an ambiguous and multifaceted concept, and he began to wonder whether he had “sent himself on a fool’s errand.”¹³ The subtitle of his paper, “I Fought the Topoi and the Topoi Won” suggests the difficulty of the line of research that he pursued.

Chapter 8 of Walton, Reed and Macagno’s book gives an account of the history of argumentation schemes that links schemes to the historical study of topics, commenting that many have interpreted the topic as a device to help an arguer search around to find an argument that could be useful for example in a debate or in a court of law.¹⁴ This approach suggests a way of approaching the

¹¹ Macagno and Walton, “Classifying the Patterns of Natural Arguments”, *supra* n. 2.

¹² Michael Leff, “The Topics of Argumentative Invention in Latin Rhetorical Theory from Cicero to Boethius” (1983) 1(1) *Rhetorica: A Journal of the History of Rhetoric* 23-44, p. 23.

¹³ Michael Leff, “Up from Theory: Or I Fought the Topoi and the Topoi Won” (2006) 36(2) *Rhetoric Society Quarterly* 203-211.

¹⁴ Douglas Walton, Christopher Reed, and Fabrizio Macagno, *Argumentation Schemes* (Cambridge: Cambridge University Press, 2008).

topoi by seeing them as argument sources from which the individual arguments are instances, and templates from which many individual arguments can be constructed.¹⁵ This direction readily suggests the approach of equating the Aristotelian *topoi* with the kinds of argumentation schemes listed in the compendium of schemes presented in Chapter 9 of the same book.¹⁶ How well this hypothesis proves to be sustainable needs further investigation, but there is some evidence in favour of it in Cicero's *De Inventione*.

In *De Inventione* Cicero lists and defines a list of concepts that were centrally important to legal argumentation in his day, and continue to be in our day.¹⁷ These include the following notions: habit, feeling, interest, purpose, time, opportunity, manner, referring to the state of mind in which an act was performed, facilities and conditions which make something easier to do, and the consequence. In *De Inventione*¹⁸ he writes that all argumentation (*argumentatio*) drawn from these topics (*loci*) as indicated in this list have to be either probable or irrefutable. This way of expressing the connection between topics in argumentation is noteworthy, because it suggests that arguments are drawn from the topics. This makes one wonder whether the topics themselves can be seen as having the form of arguments, comparable to argumentation schemes, or whether the topics are something different (places?) from which the arguments are drawn. Cicero's descriptions of the concepts he classifies as topics in his list are hard to grasp in any precise way, and this is perhaps the reason why it has been so difficult for scholars after the ancient world to make any practical use of the topics. What can help here is to give an example that Cicero offered as

¹⁵ Macagno and Walton, "Classifying the Patterns of Natural Arguments", *supra* n. 2.

¹⁶ Walton, Reed and Macagno, *supra* n. 14.

¹⁷ Cicero, *supra* n. 5, I, 6, pp. 35-43.

¹⁸ *Ibid.*, p. 44.

representing the type of argumentation used in a typical criminal case of the kinds he encountered as a practicing lawyer.

Cicero, in *De Inventione*¹⁹ outlines a typical criminal case of the kind he was familiar with, concerning the following story. A traveller fell into companionship with another man who was on a business trip and was carrying a considerable sum of money. Stopping at the same inn, they planned to share an apartment. After dinner they slept in the same room of the apartment where they fell into a deep sleep. During the night while the two men were asleep, the innkeeper took some money which was on the bed of the one man, and taking his sword, which was also on the bed, killed the other man with it. He then put the blood-stained sword back into its sheath. When the innkeeper entered the room in the morning, he found the one man dead and other gone. After the innkeeper made the allegation of murder, some guests who pursued the traveller drew his sword and found it stained with blood. The charge made was that this man had committed murder, and his answer to the charge was that he did not. The other man claimed that in the morning, he had called his companion to get up, but hearing no answer, took his sword along with the rest of his belongings and set out alone. Cicero²⁰ writes that from these facts rises the central issue which he calls the issue of fact, the question of whether the accused man committed murder. Cicero does not tell us the outcome of the trial, but he does tell us that the truth was found out when the innkeeper had been caught in a different crime.

From this point onward, Cicero proceeds to explain the arguments on both sides. What he calls the cause of an act²¹ falls under the head of impulse and premeditation. Impulse is what urges a person to do something without thinking

¹⁹ *Ibid.*, II, 4, pp. 14-16.

²⁰ *Ibid.*, II, 4, p. 15.

²¹ *Ibid.*, II, 4, p. 20.

about it, whereas premeditation is careful and thoughtful reasoning about doing or not doing something. Cicero writes²² that the topic is the foundation or basis of the issue which seeks out why the act was done according to some reason. In a typical case, the defence will say that the act was done on impulse, whereas the prosecutor will say that the defendant carried out the action deliberately in order to obtain some advantage or avoid some disadvantage.²³

So here we come back to the question of what topics were involved in the argumentation in this typical case. Cicero added²⁴ that a familiar line of argument under this topic is for the prosecutor to argue that no one else had a motive for committing the crime. But if it seems like others might have had a motive, it must be shown that they lack the power, the opportunity or the desire. The counsel for the defence will maintain that there was no impulse, or if there was one he will try to prove that it was only a weak emotion or one from which this kind of deed does not generally arise.²⁵ He will weaken the suspicion of premeditation if he says that there was no gain for the defendant, or that there was greater gain for others.

What Cicero's remarks suggest is that topics are general patterns or templates representing species of argumentation that play a role in a typical a case where there is a central issue, such as whether the defendant committed murder or not, and there are some standard types of pro and con arguments used to represent both sides in the argumentation that is put forward in the trial. One can see a similarity here between certain argumentation schemes and the general patterns representing species of argumentation described by Cicero. There are quite a few schemes of this sort mentioned, but here two will be identified.

²² *Ibid.*, II, 4, p. 19.

²³ *Ibid.*, II, 4, p. 20.

²⁴ *Ibid.*, II, 4, p. 20.

²⁵ *Ibid.*, II, 4, p. 23.

The first is the argumentation scheme for argument from motive to action.

- *Conditional Premise*: If agent *a* had a motive to bring about action *A* then *a* is somewhat more likely to have brought about *A* than another agent who lacked a motive.
- *Motive Premise*: *a* had a motive to bring about *A*.
- *Conclusion*: *a* is somewhat more likely to have brought about *A*.

This form of inference was structured by Leonard²⁶ as a form of argument with two premises and a conclusion and modelled as an argumentation scheme for argument from motive to action by Walton.²⁷ So far then, we have seen how an argument that goes from a motive to an action can be configured with this argumentation scheme.

The second is the argumentation scheme for argument from evidence to motive.

- *Conditional Premise*: If there is evidence of agent *a*'s actions or statements indicating that *a* had a motive to bring about action *A* then *a* had a motive to have brought about *A*.
- *Evidence Premise*: There is evidence of agent *a*'s actions or statements indicating that *a* had a motive to bring about action *A*.
- *Conclusion*: *a* had a motive to have brought about *A*.

Idaho v Davis,²⁸ which concerned the struggle between sheepherders and cattlemen to control land, furnishes examples of the use of both schemes in a trial.

²⁶ David Leonard, "Character and Motive in Evidence Law" (2001) 34 *Loyola of Los Angeles Law Review* 439-536, p. 470.

²⁷ Douglas Walton, "Teleological Argumentation to and from Motives" (2011) 10(3) *Law, Probability and Risk* 203-223, p. 205.

²⁸ *State v Davis*, [1898] 6 Idaho 159, 53 P. 678 in Leonard 2001, p. 470.

In order to understand how argument diagrams work, it is necessary to draw a distinction between linked and convergent arguments. In a linked argument, there is more than one premise, and the premises function together to give support to the conclusion. A typical linked argument has two premises and it is clear that the two premises function together to support the conclusion because the given argument fits an argumentation scheme of the type that has two premises. In a convergent argument, there are also two or more premises but each premise (or each group of premises) function together to support the conclusion.

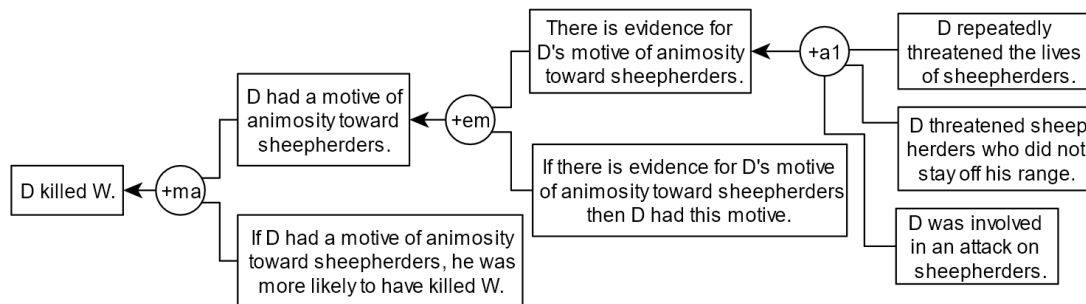


Figure 1: Argument Diagram of Arguments to and from Motive

The argument diagram shown in figure 1 was drawn using the conventions of the Carneades Argumentation System, where the ultimate conclusion appears in the leftmost rectangle and the names of the argumentation schemes are given in the round argument nodes. The notation *ma* represents the scheme for argument from motive to action and the notation *em* represents the scheme for argument from motive. The plus in the node denotes a pro argument, an argument that provides positive support for its conclusion. There are three linked arguments. The two on the left are easily seen to be linked, because each of them fits an argumentation scheme that has two premises. The one on the right does not fit any known scheme, but on the assumption that its three premises go together to support the conclusion, *a₁* can be represented as a linked argument.

Walton and Schafer²⁹ used more complex examples of legal argumentation to show how schemes for argument from motive to action and argument from evidence to motive are combined with other argumentation schemes, such as the ones for practical reasoning and abductive reasoning (inference to the best explanation) in larger argument diagrams.

3 The Watson Debater tool

It is well-known that IBM's Watson can answer factual questions by extracting information from a database of natural language texts, of the kind familiar from the TV game show *Jeopardy*. In 2014, IBM demonstrated a program called Debater that employs some of the text processing technology of the Watson program to perform argument invention.

Watson Debater is a computational tool to assist a human user to find pro or con arguments in relation to an issue being discussed. Debater uses the word "topic" in a technical sense, defining a topic as a short statement that poses an issue such as whether the sale of violent video games to minors should be banned.³⁰ The user inputs a topic and then Debater helps the user find what is called a context dependent claim (CDC), a statement that directly supports or contests a topic. So Debater finds pro or con arguments by finding CDC's that can be used as premises in arguments on either side of the topic. It carries out this task by using a variety of search engines.³¹ The *topic analysis engine* is used to

²⁹ Douglas Walton and Burkhard Schafer, "Arthur, George and the Mystery of the Missing Motive: Towards a Theory of Evidentiary Reasoning about Motives" (2006) 4(2) *International Commentary on Evidence* 1-47.

³⁰ Ran Levy et al., "Context Dependent Claim Detection" (2014) *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)* 1489-1500, available at <http://www.aclweb.org/anthology/C/C14/C14-1141.pdf> (accessed 17 December 2014).

³¹ Ehud Aharoni et al., "Claims on Demand — An Initial Demonstration of a System for Automatic Detection and Polarity Identification of Context Dependent Claims in Massive Corpora" (2014) *Proceedings of the 25th International Conference on Computational Linguistics*

identify the main concepts mentioned in a topic and the sentiment towards each of these concepts. The *article retrieval engine* searches for Wikipedia articles that have a high probability of containing CDC's. The *CDC detection engine* zooms in within the retrieved articles to detect CDC's. The *CDC pro/con engine* automatically judges the polarity of a CDC found, with respect to a given topic. IBM developed and tested Debater by training a team of human labellers to search for CDC's in a selected collection of Wikipedia articles. This team of human labellers identified and marked up CDC's in a selected collection of Wikipedia articles.

In a video demonstration of Debater, when the proposition "The sale of violent videogames to minors should be banned" was selected as a topic, Debater collected the strongest pro and con arguments. The outcome that was produced can be represented visually in the Carneades style of argument diagram shown in figure 2. As the reader will recall from section 2, a convergent argument is one where each premise independently supports the conclusion.

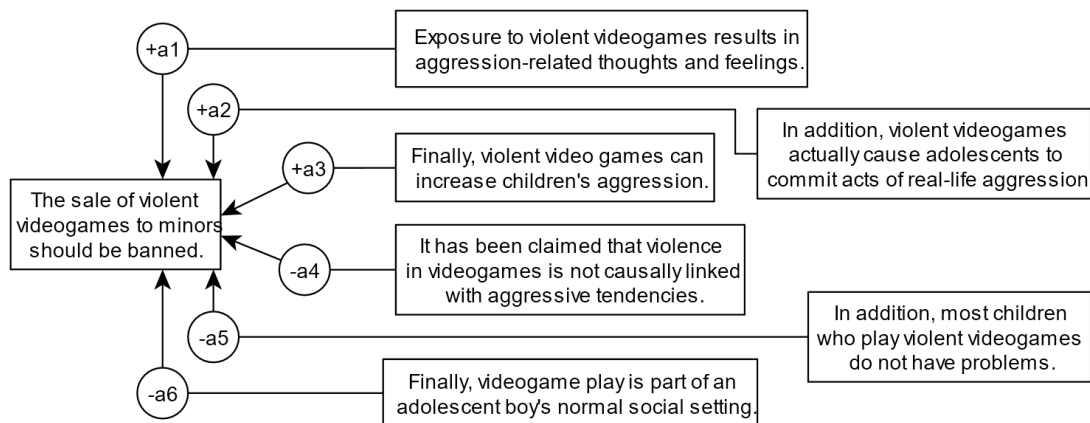


Figure 2: Leading Arguments Found by Debater on the Videogames Topic

In figure 2, six arguments are shown, and each of them is represented as a separate argument. Hence the overall structure of the argumentation in this case is that of a convergent argument. Wikipedia produced a large number of candidate CDC's initially, but they had to be narrowed down to a selected number of them that would be useful to an arguer who wants to find usable arguments on the pro or con side of the topic. The non-useful ones have to be filtered out. For example, the statement that violent video games can increase children's aggression was selected as a CDC, but the statement that violent video games should not be sold to children, also found as a CDC, had to be excluded, because it merely restates the topic. In another example,³² the CDC claiming that violence in games hardens children to unethical acts was included in a longer sentence stating that two named individuals argue that violence in games hardens children to unethical acts, and goes on to call first-person shooter games murder simulators. In this instance the CDC was contained in the middle of a long sentence containing several arguments. The task here is finding the boundaries of the text that contains the CDC enabling the exclusion of parts of the sentence that are not useful.

Speed is taken to be important for a device of this sort. The CDC's have to be produced shortly after the user inputs the topic, and then Debater makes them available to the user in a voice format. Lippi and Torroni³³ however classify the Watson Debater as a system of argument mining. Ashley also sees Debater as an argument mining tool.³⁴

³² Levy et al., *supra* n. 30, p. 1489.

³³ Marco Lippi and Paolo Torroni, "Argument Mining: State of the Art and Emerging Trends" (2016) 16(2) *ACM Transactions on Internet Technology* 10:1-10:25, p. 10:11.

³⁴ Ashley, *supra* n. 3, p. 23.

4 Argument mining

Mochales and Moens³⁵ used argument mining to annotate a set of legal documents containing judges' legal decisions extracted from a database of cases in the European Court of Human Rights. Argumentation schemes were applied to the task of identifying arguments in the text, using discourse indicators such as "it follows that" and "in conclusion." Other indicators of commonly used terms in this type of document, such as "in the view of the factfinder," were used to identify premises of arguments. Argument mining was greatly assisted by the structure of the way the database is organised in the European Court of Human Rights. The sections selected to compile the corpus contained only summaries of the judges' arguments used to support their conclusions.

From an argumentation point of view, the main goal of argument mining is to build an automated technology that can be used as a tool to search through a natural language text and identify the arguments (pro or con claims) in it, and their parts (premises and conclusions), or alternatively to help human coders to carry out this task.³⁶ But research efforts on argument mining in computational linguistics have expanded rapidly since 2011, producing eleven distinct methods for argument mining that have been developed since then.³⁷ For this reason, Lippi and Torroni³⁸ have expressed the main goal of argument mining from a point of view of computer science in a different way: "the main goal of argumentation mining is to automatically extract arguments from generic textual corpora, in order to provide structured data for computational models of argument and reasoning engines." Here the term "argumentation mining" is

³⁵ Raquel Mochales and Marie-Francine Moens, "Argumentation Mining" (2011) 19(1) *Artificial Intelligence and Law*, pp. 1-22.

³⁶ Walton, *supra* n. 27.

³⁷ Lippi and Torroni, *supra* n. 33, p. 10:11.

³⁸ *Ibid.*, p. 10:2.

used, but it can be presumed to be equivalent to the more commonly used term in argumentation studies, “argument mining”.

Three main approaches to argument mining have been compared by Lawrence and Reed.³⁹ The discourse indicators approach uses verbal indicators in a natural language text such as “therefore” and so forth, that point to the occurrence of an argument, along with its components, its premises and conclusion. The topical similarity approach studies how changes in the topic of a discussion relate to the argumentation structure in the text. The supervised machine learning approach is based on argumentation schemes that enable the identification of premises and conclusions and show how these components work together as parts of an argument.

Verbal indicators of the kind used in the discourse indicators approach are terms such as “therefore”, “because”, “consequently”, “however”, “nonetheless”, and so forth, that indicate support for the conclusion by a set of premises, or in the case of the last two indicators, disagreement with or qualification of a prior claim. Empirical results cited in Lawrence and Reed suggest that the discourse indicators approach yields a strong indication of the connection between propositions, but the low frequency with which such verbal indicators occur in texts containing arguments suggests that they fail to help identify them in the vast majority of instances.⁴⁰

The topical similarity approach represents the argument structure in a given case as a tree where the conclusion, the root of the tree, is given first and the line of reasoning is followed supporting this conclusion. When that line of

³⁹ John Lawrence and Christopher Reed, “Combining Argument Mining Techniques” (2016) *Proceedings of the 2nd Workshop on Argumentation Mining* 127-136, available at <http://www.aclweb.org/anthology/W15-0516> (accessed 1 November 2017).

⁴⁰ *Ibid.*, p. 129.

reasoning is exhausted, the argument goes back up the tree to investigate for further support.⁴¹

The supervised machine learning approach splits text into sentences so that features of each sentence can be used to classify them into the category of argument or the category of non-argument. But beyond this, once the components of a given argument found in a text have been identified, the argument can be fitted to the requirements of a specific argumentation scheme representing a type of argument that is included in a list of specific known schemes, such as that of Walton, Reed and Macagno.⁴² This approach is represented by the methodology of Feng and Hirst.⁴³ They used the sixty-five schemes of Walton et al.⁴⁴ and interestingly found that the number of instances of five of these schemes made up 61% of the arguments identified in their database.⁴⁵ These five schemes were the ones for: argument from example, argument from cause to effect, practical reasoning, argument from consequences and argument from verbal classification.

Lawrence and Reed⁴⁶ showed how these three methods can be used in combination to achieve results that are close to the analysis of the text by human coders. These results suggest that a combined approach yields much better results and performance than any single approach.

Argument invention often appears to be based on the same techniques used in argument mining, raising the questions of whether the two methods are

⁴¹ *Ibid.*, 129.

⁴² Walton, Reed and Macagno, *supra* n. 14.

⁴³ Vanessa Feng and Graeme Hirst, "Classifying Arguments by Scheme" (2011) *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* 987-996, available at <http://www.aclweb.org/anthology/P11-1099> (accessed 1 November 2017).

⁴⁴ Walton, Reed and Macagno, *supra* n. 14.

⁴⁵ Feng and Hirst, *supra* n. 43, p. 988.

⁴⁶ Lawrence and Reed, *supra* n. 39.

different, and if so how they differ. The basic difference resides in the goal of each method as a practical tool. The purpose of argument invention is to help a user find arguments to support or attack a designated claim, a particular proposition selected by the user at the outset. The basic purpose of argument mining is to begin with a natural language text taken to contain arguments, and search through it to identify instances of arguments and their parts.⁴⁷

Lippi and Torroni⁴⁸ represent the typical argument mining system architecture as a pipeline taking the user from the raw text of an unstructured document to produce the output of a structured document where the detected arguments and the relations are annotated in the form of an argument graph. We consider argument diagrams, or maps, to be visualisations of argument graphs, where the graphs are the underlying mathematical or logical structure of the arguments. Seeing an argumentation mining system as having a structure of this kind shows how naturally it would be to use an argument diagram to annotate the output of the pipeline.

Although Watson is proprietary, it is based on an open source text processing tool, the Unstructured Information Management Architecture (UIMA). It is not known yet exactly how Debater or UIMA will be applied to legal argumentation,⁴⁹ but to appreciate the possibilities of new advances in this area, it is useful to consider some of the leading computational models of legal argumentation that have been built in AI and law research.

⁴⁷ Andreas Peldszus and Manfred Stede, "From Argument Diagrams to Argumentation Mining in Texts: A Survey" (2013) 7(1) *International Journal of Cognitive Informatics and Natural Intelligence* 1–31.

⁴⁸ Lippi and Torroni, *supra* n. 33, p. 10:5.

⁴⁹ Ashley, *supra* n. 3, p. 26.

5 Computational argumentation systems

The formal argumentation system ASPIC+ is based on a set of strict and defeasible inference rules expressed in a logical language L . A knowledge base K consists of a set of propositions that can be used along with the inference rules to generate arguments.⁵⁰ Arguments take the form of trees containing (1) nodes representing propositions from L , and (2) edges from a set of nodes $\varphi_1, \dots, \varphi_n$ to a node ψ making up an argument from premises $\varphi_1, \dots, \varphi_n$ to a conclusion ψ .

ASPIC+⁵¹ evaluates arguments by means of applying abstract argumentation frameworks.⁵² In such a framework, arguments are evaluated on the basis of attack relations among arguments. The resulting argumentation is modelled using a graph structure representing attack relations of this kind: a_1 attacks a_2 , a_2 attacks a_3 , a_3 attacks a_2 , and a_2 attacks a_1 . An argument can be *in* (accepted) or *out* (defeated). An argument is *out* if it is attacked by any other argument that is *in*. An argument is *in* if there is no successful (*in*) argument attacking it. The system ASPIC+ uses defeasible argumentation schemes, such as DMP (defeasible modus ponens), but it can also use the deductive form of *modus ponens*. ASPIC+, along with other systems explained below, offers three ways of attacking an argument: attacking a premise, attacking the conclusion or attacking the inferential link between the premises and the conclusion. The last type of attack is called an undercutter.

⁵⁰ Sanjay Modgil and Henry Prakken, "The ASPIC+ Framework for Structured Argumentation: A Tutorial" (2014) 5(1) *Argument & Computation* 31-62.

⁵¹ Henry Prakken and Giovanni Sartor, "Argument-based Extended Logic Programming with Defeasible Priorities" (1997) 7 *Journal of Applied Non-classical Logics* 25-75.

⁵² Phan Minh Dung, "On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-person Games" (1995) 77(2) *Artificial Intelligence* 321-357.

The formal argumentation system DefLog⁵³ is based on two primitive notions, dialectical negation and defeasible implication.⁵⁴ Dialectical negation represents the defeat of an argument. Arguments can be justified or defeated.⁵⁵ The notion of one argument a_x defeating another argument a_y is modelled as a rebutting defeater in Pollock's⁵⁶ sense, meaning that a_x defeasibly implies the dialectical negation of a_y . To qualify as justified, an argument must not be defeated by an argument having justified statements as premises. DefLog has an automated argument assistant ArguMed that assists a user to construct an argument diagram to analyse and evaluate arguments.⁵⁷

The Carneades Argumentation System⁵⁸ was named after the Greek philosopher Carneades who had a fallibilistic theory of knowledge based on defeasible reasoning.⁵⁹ Carneades (the system) models arguments as directed graphs consisting of two kinds of nodes. Statement nodes contain statements (propositions) that function as premises or conclusions in arguments. Argument nodes join premises to conclusions. The argument nodes can contain a variety of deductive or defeasible argumentation schemes. Argument nodes are of two kinds. A pro argument supports a proposition. A con argument attacks a proposition. An argument graph is visually displayed as an argument map. Carneades follows ASPIC+ in modelling the three kinds of argument attacks.

⁵³ Bart Verheij, "Argumentation Support Software: Boxes-and-Arrows and Beyond" (2007) 6 *Law, Probability and Risk* 187-208.

⁵⁴ Bart Verheij "DefLog: on the Logical Interpretation of Prima Facie Justified Assumptions" (2003) 13(3) *Journal of Logic and Computation* 319-346.

⁵⁵ Verheij, "Argumentation Support Software: Boxes-and-Arrows and Beyond", *supra* n. 53.

⁵⁶ John Pollock, *Cognitive Carpentry* (Cambridge, Mass.: The MIT Press, 1995).

⁵⁷ Bart Verheij, "Automated Argument Assistance" (2002), available at <http://www.ai.rug.nl/~verheij/aaa/argumed3.htm> (accessed 1 November 2017).

⁵⁸ Thomas Gordon, "The Carneades Argumentation Support System" in Christopher Reed and Christopher Tindale (eds.), *Dialectics, Dialogue and Argumentation* (London: College Publications, 2010), pp. 145-156.

⁵⁹ Harald Thorsrud, "Cicero on his Academic Predecessors: The Fallibilism of Arcesilaus and Carneades" (2002) 40(1) *Journal of the History of Philosophy* 1-18.

In Carneades, argument graphs are evaluated by assuming that an audience determines whether the premises of an argument are accepted or not, and argument weights (fractions between zero and one) can be assigned to each argument, representing the strength of the audience's acceptance. Carneades evaluates arguments by calculating whether the conclusion should be accepted based on acceptance of the premises and on the argumentation scheme that forms the link joining the premises to the conclusion. An argument is said to be applicable if all its premises are accepted by the audience. Conflicts between pro and con arguments are resolved using proof standards, such as preponderance of the evidence or clear and convincing evidence.⁶⁰ The proof standards are not defined numerically, but using thresholds α and β , as follows:⁶¹ The preponderance of the evidence standard for a proposition p is met if and only if there is at least one applicable argument pro p , and the maximum weight assigned by the audience to the applicable arguments pro p is greater than the maximum weight of the applicable arguments con p . The clear and convincing evidence standard is met if and only if (1) the preponderance of the evidence standard is met, (2) the maximum weight of the applicable pro arguments exceeds some threshold α , and (3) the difference between the maximum weight of the applicable pro arguments and the maximum weight of the applicable con arguments exceeds some threshold β .

There are currently four successive versions of Carneades that have been implemented, which are all accessible online.⁶² ASPIC+, DefLog and all four

⁶⁰ Thomas Gordon and Douglas Walton, "Proof Burdens and Standards" in Iyad Rahwan and Guillermo Simari (eds.), *Argumentation and Artificial Intelligence* (Berlin: Springer, 2009), pp. 239-260.

⁶¹ *Ibid.*, p. 245.

⁶² Thomas Gordon, "Carneades Argumentation System" (2017), available at <https://carneades.github.io/> (accessed 1 November 2017).

systems of Carneades assume a distinction between three kinds of cases where one argument attacks another. An argument can attack a premise of another argument, or it can attack its conclusion. Or it can attack the inferential link between the premises and the conclusion. ASPIC+ can use value-based reasoning to break deadlocks between arguments.⁶³ A deadlock occurs where argument a_x attacks argument a_y but a_y also attacks a_x . In ASPIC+ the deadlock can be resolved, showing which side has the stronger argument, if there was a priority ordering of values. The argument based on the higher priority value wins. All three systems can use value-based reasoning.

DefLog and ASPIC+ can assist a user in the task of argument construction with the aid of knowledge-based systems and argument mapping devices. The knowledge bases used to construct arguments in DefLog and ASPIC+ are propositional rules. They are comparable to the argument nodes in Carneades argument graphs. That is, a knowledge base in Deflog and ASPIC+ serves the same function as an argument graph in Carneades. But, as the examples in the rest of this paper will show, by argument invention or construction, we mean the process of finding and adding further arguments to the argument graph. Neither ASPIC+ nor Deflog can do this. The knowledge base in Deflog and ASPIC+ functions as the argument graph, and so the argument graph is static. Neither ASPIC+ nor Deflog provides any way to dynamically construct new arguments extending the argument graph. Carneades is different. In addition to the argument graph, Carneades also provides “theories” consisting of a set of argumentation schemes. Argumentation schemes, like inference rules, are abstract arguments containing schema variables. Arguments are constructed by instantiating schemes, substituting schema variables with constant terms.

⁶³ Modgil and Prakken, *supra* n. 50.

Arguments are found and constructed by using an inference engine along with a heuristic search strategy to construct arguments by instantiating the schemes, starting with the “facts” accepted or assumed to be accepted by the audience.

Despite the capability of these systems to model argument construction as well as argument evaluation, so far, both in the AI literature and in argumentation studies generally, the direction of research has been almost exclusively on argument evaluation rather than on argument invention. This direction is understandable, given the traditional focus of logic on the task of argument evaluation. But at this point, we feel that it is important to draw attention to the new resources for argument invention as well, given its importance for both logic and rhetoric, and for its showing how closely the two tasks are connected. This connection is remarkable in an era when the two fields are taken to be so separate, and where there is even a longstanding hostility between them, from Plato onwards.

6 Argument invention in computational systems

Building a computational argument invention system that can be used for practical purposes to assist an arguer to search for and find arguments that the arguer can use to persuade an audience to come to accept some proposition requires a general framework. Using the Carneades argument assistant, the rhetorical persuader addresses an audience, and she has some idea about propositions in the commitments of the audience, the propositions already accepted by the audience.

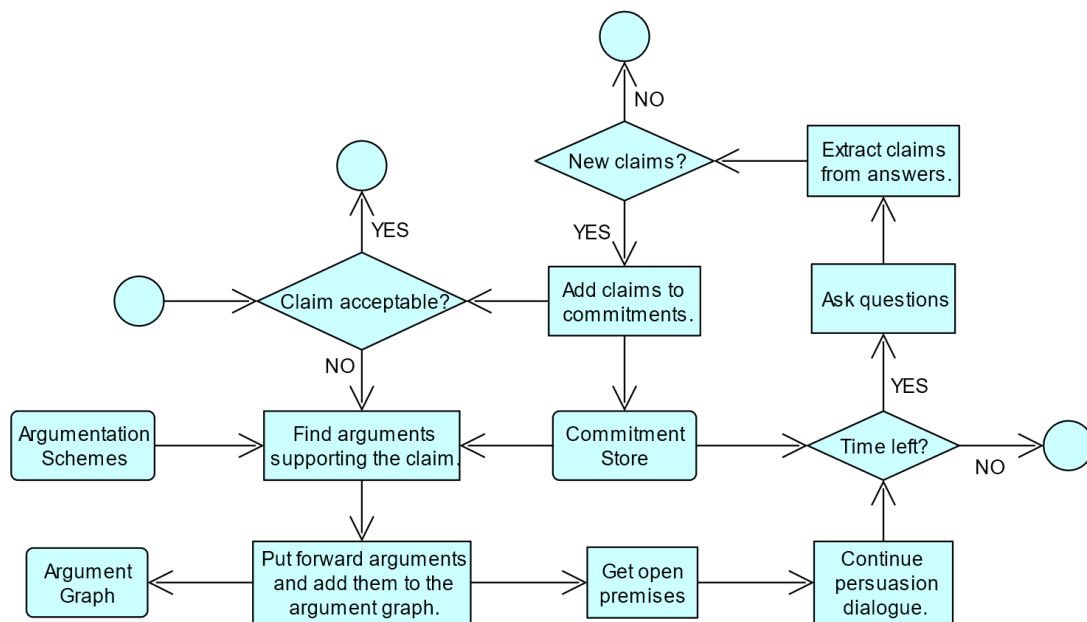


Figure 3: User Activities in the Carneades Invention System

The Carneades argument assistant, as shown in figure 3, is being used in a persuasion dialogue.⁶⁴ The assistant constructs a chain of argumentation where the conclusion of the chain is the goal proposition. That is, the goal is the proposition that the speaker wants to get the audience to accept, and the assistant uses backward reasoning to track from this goal to collect premises that can be used to prove it. This goal proposition is called the arguer’s ultimate claim, or ultimate probandum, the endpoint of the chain of argumentation representing the proposition to be proved, as described by the ancient status theory. The argument assistant searches through the commitments of the audience and uses argumentation schemes in its knowledge base to construct an argument to prove the ultimate claim.

⁶⁴ Henry Prakken, “Formal Systems for Persuasion Dialogue” (2006) 21(2) *The Knowledge Engineering Review* 163-188.

An outline of how the arguer's task needs to proceed as aided by the system is displayed visually in the process model outlined in figure 3. What the argument assistant needs to do within the constraints of this framework is to build a sequence of argumentation by applying schemes in the knowledge base to derive the ultimate claim, based on arguments that the audience will accept, and on premises that are either already accepted by the audience or that can be derived from accepted premises by arguments that the audience accepts.

The Carneades argument assistant can apply the schemes from the given repository of schemes to the set of available premises in the knowledge base and use them to generate new arguments. Carneades can do this automatically. All the user (arguer) needs to do is to ask the automated assistant to apply the schemes to find arguments. This is possible because Carneades has an inference engine for applying argumentation schemes to construct arguments. In Carneades 1.0.2, the schemes are applied to the accepted and rejected statements in the argument graph. These statements are used as "facts" by the inference engine. Version 3 of Carneades does this as well, but goes further by including a dialogue component to interactively ask the user for additional facts during the search for arguments.⁶⁵ If these arguments prove the conclusion, the sequence can stop because the search has been successful. But if success has not been achieved yet, the automated argument assistant can find potential arguments that go part of the way towards proving the conclusion but still have gaps. These

⁶⁵ Carneades 4 does not have a dialogue component for asking users to enter facts interactively. This is because Carneades 3 uses backwards-chaining, in a goal-directed way, whereas Carneades 4 uses forwards-reasoning to derive arguments from argumentation schemes and assumptions. Both strategies, forwards and backwards reasoning, have their advantages. Forwards reasoning allows us to invent arguments using argumentation schemes, like Argument from Expert Witness Testimony, where the conclusion is a second-order variable ranging over propositions. That is, only Carneades 4 can construct arguments using formalisations of the main twenty schemes.

gaps are propositions that the audience does not accept, but if they did accept them, they could fit into sequences of argumentation that would prove the ultimate conclusion.

At this point it is important to recognise that there are three kinds of situations that might be encountered. In the first kind of situation, the speaker may be presenting a televised message, or writing a speech of a kind that does not allow interaction with the audience. In the second kind of situation, the speaker may be able to interact with a live audience, or to communicate with the audience by means of a device such as the Internet. In this kind of case, as shown at the bottom of figure 3, the speaker can try to persuade the audience to accept new propositions that might be useful as parts of the sequence of argumentation needed for moving towards filling the gaps required to prove the ultimate conclusion. She might be able to ask questions of the audience, and use the answers to these questions to add to the knowledge base. Such a persuasion dialogue can continue until time or costs prevent further dialogue and collection of new knowledge. In the third kind of case, two speakers are addressing a third-party audience in a debate format. One speaker has a thesis that she is trying to persuade the audience to come to accept, while the other speaker is trying to persuade the audience to come to accept a proposition that is the opposite of the thesis of the first speaker.

At this point the reader should be warned to be careful to distinguish between formal computational models of argumentation and software tools. There are many software tools for helping a user to make argument diagrams, but such a tool by itself does not represent a computational model. Also, some but not all of the computational models discussed in this paper have this kind of tool for drawing argument diagrams. In discussing argument invention, Carneades has mainly been used for purposes of illustration, because it is a formal and computational argumentation model that also has a visualisation tool

that can be used to help make argument diagrams. It is one of the few systems with an inference engine for inventing arguments by instantiating argumentation schemes.

7 Argument invention using Carneades

A small example, illustrated in figure 4, can be used to explain how the Carneades method of argument invention works. Literal propositions p_0, p_1, \dots, p_n are shown in boxes. A literal proposition is a simple proposition that contains no conjunctions, disjunctions or conditionals. By convention, only positive literals are displayed in the boxes. To represent negation, a con argument can be used. The propositions in the boxes function as premises or conclusions of the arguments. A proposition that has been accepted by the audience is shown in a box with a green background. A proposition that has been rejected by the audience is shown in a box with a red background. A proposition that has neither been accepted nor rejected is shown in a box with a white background. The arguments a_1, \dots, a_n are shown in circles. An argument that has a plus sign in front of its proposition is a pro argument. An argument that has a minus sign in front of its proposition is a con argument. For example, the notation $+a_1$ represents the first pro argument. Information about the argumentation scheme fitting the argument is contained in the circle, although this feature is not shown on the diagram in figure 4.

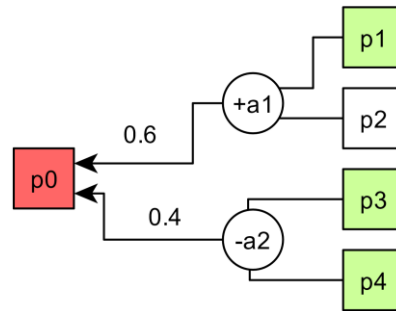


Figure 4: First Argument Construction Step in the Example

The ultimate proposition to be proved by the speaker, p_0 , is shown at the far left of figure 4. There are two arguments directly concluding to p_0 , namely a_1 and a_2 . a_1 is a pro argument, as indicated by the plus sign in its argument node. a_2 is a con argument, as indicated by the minus sign in its argument node. Carneades models the audience as a set of accepted and rejected statements and an assignment of weights to arguments. The accepted and rejected statements in the model of the audience are used as “facts” when applying argumentation schemes in a theory to find and construct arguments. The argument assistant looks around in the knowledge base and finds that p_1 is accepted by the audience, but p_2 is not accepted. This outcome means that argument a_1 does not prove the conclusion p_0 . In order for it to prove the conclusion p_0 , both premises p_1 and p_2 have to be accepted.

When the argument assistant searches through the knowledge base of the audience, it finds that both p_3 and p_4 are accepted. So at this point in the development of the argument, the outcome is not good news for the speaker. Since both premises of the con argument are accepted, this argument is applicable, meaning that the argument a_2 defeats the conclusion p_0 that the speaker is supposed to prove. In other words, on the total body of evidence so far, Carneades calculates that the proposition p_0 , should be rejected. On the

preponderance of evidence proof standard there is one applicable argument with weight 0.4 attacking p_0 , and one argument with weight 0.6 supporting the argument, but is ineffective because it is not applicable.

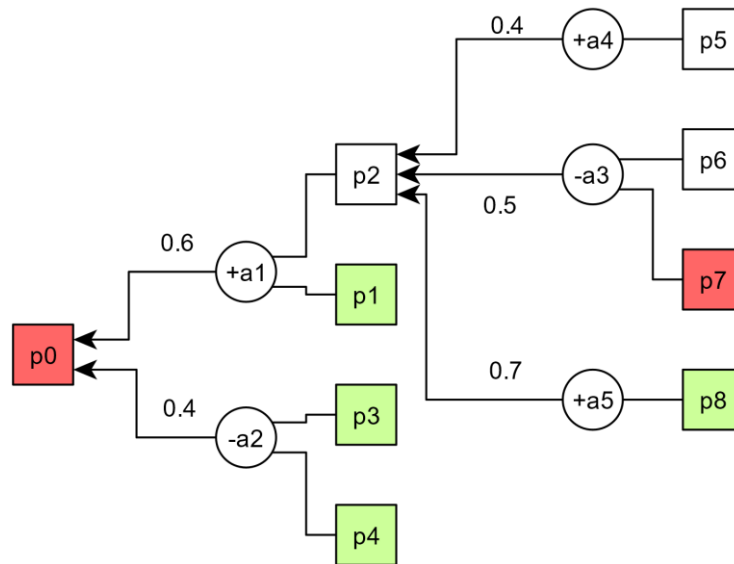


Figure 5: Second Argument Invention Step in the Example

The speaker still wants to persuade the audience to accept p_0 , and so she asks the argument assistant to provide some help in building an argument that could do this. The argument assistant checks the knowledge base and verifies that proposition p_1 is accepted by the audience, but proposition p_2 is not accepted. So now what the speaker has to do is to find some way to persuade the audience to accept p_2 . Or if there are already propositions in the knowledge base of the audience that could be used in arguments to attack p_2 , the assistant should also search for some arguments the speaker can use to defend against these attacks.

Let us suppose the next thing the argument assistant finds is that there is a con argument, a_3 , that could be fitted to two premises p_6 and p_7 , so that a_3 could be used to attack p_2 . But searching through the knowledge base of the audience, the assistant finds that p_6 is neither accepted nor rejected by the audience, and p_7

is rejected by the audience. On this basis the argument assistant tells the speaker not to worry about this argument.

The argument assistant searches again in the knowledge base of the audience and finds that there is a pro argument a_4 that could be used to support acceptance of p_2 , except that the only premise available for this argument is neither accepted nor rejected by the audience. So one recommendation the argument assistant brings forward to the speaker is that she could look for some further arguments to support p_5 .

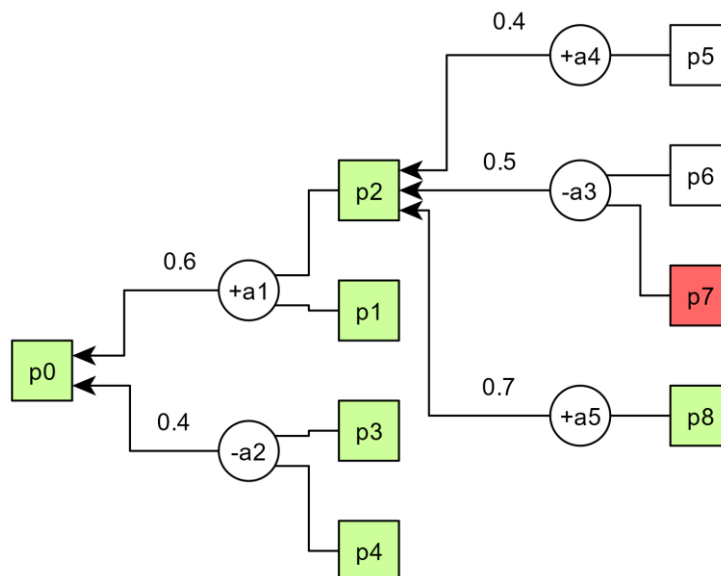


Figure 6: Third Argument Construction Step in the Example

But there is also an even easier solution available to the problem. The assistant finds another argument a_5 that has only a single premise p_8 , and that premise is accepted by the audience. On this basis, Carneades calculates that a_5 proves that proposition p_2 is accepted by the audience, and therefore in the argument map displayed on the computer screen Carneades would automatically show p_2 with a green background. This outcome is shown in figure 6. Now, as shown in figure 6, both premises of the pro argument a_1 are displayed

in green boxes, showing that this argument is applicable. Therefore, Carneades automatically shows p_0 in a green box instead of a red box.

To sum up what has happened, in the initial state of the argument, the con argument a_2 defeated the speaker's ultimate claim to be proved, p_0 . But following the advice of the argument assistant, the speaker was able to find a pro argument a_1 with weight 0.6, to attack the opposing argument a_2 which has weight 0.4. Let us say that the proof standard is that of the balance of the probabilities. This means that the argument that is stronger for the audience will win out over the argument that is weaker for the audience, even if it is only slightly weaker. Therefore, the argument assistant has provided a means that the speaker can use to prove her ultimate thesis by finding a pro argument that defeats the existing con argument that was initially posed to attack her thesis.

8 Argument invention in a debate framework

In this section it is shown how Carneades can be applied to the task of argument invention in the third type of situation. This type of situation is more complex than the first two, because it has the structure of a debate in which one party is pro the designated proposition while the other party is con. Also, each party is assumed to have its own knowledge base from which it can draw arguments directed against the arguments of the other side.

The first speaker's goal in the debate is to persuade the audience that Wikipedia is an unreliable source. Using her knowledge base, she finds some arguments and puts forward the argumentation shown in figure 7, displayed in the Carneades style.

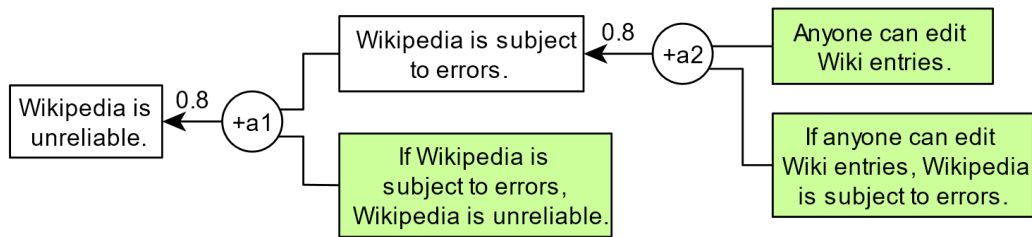


Figure 7: First Speaker's Argument in the Wikipedia Example

The ultimate conclusion, the statement that Wikipedia is unreliable, is shown at the far left. The first pro argument has two premises, the proposition that Wikipedia is subject to errors and the proposition that if Wikipedia is subject to errors, Wikipedia is unreliable. This argument fits the form of the DMP argumentation scheme. The former premise is supported by a second pro argument shown to the right of the first argument. This argument also has the DMP form.

Let us say that the audience finds both arguments quite strong. In figure 7, a strength value of 0.8 has been assigned to both arguments. Because the argumentation is in the form of the debate structure, the speaker who has the stronger argument will win the debate. Hence it is appropriate to assign the proof standard of the preponderance of evidence to the dialogue to determine which side wins and which side loses the debate.

Given these assumptions about the case, how does Carneades evaluate the first speaker's argument? First, look at argument a_2 in figure 7. Both premises are accepted by the audience, and the audience accepts the argument as being strong. Therefore, Carneades will automatically calculate that the conclusion "Wikipedia is subject to errors" is acceptable to the audience, showing the text box of this proposition with a green background. But once this proposition is shown as accepted by the audience, since it is one of the premises of argument a_1 , and the

other premise of this argument has been accepted, Carneades will automatically fill in a green background in the text box containing the ultimate conclusion that Wikipedia is unreliable. This will mean now that all five propositions shown in figure 8 should be coloured in green.

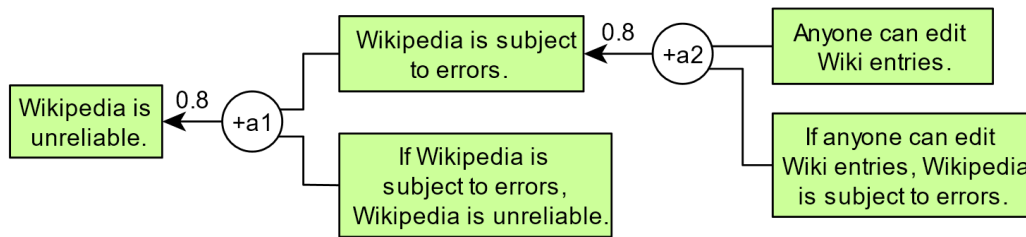


Figure 8: Second Speaker's Argument Evaluated

In short, the first speaker's argument for her ultimate conclusion that Wikipedia is unreliable easily meets its standard of proof of being more probable than not. At this point in the debate, the first speaker is winning the argument. The second speaker now needs to find a counterargument to this argument. His automated argument assistant finds in the knowledge base the proposition that a study in the journal *Nature* reported that Wikipedia is as reliable as *Encyclopaedia Britannica*. Searching around some more, the automated assistant finds that the audience accepts two other propositions as common knowledge. One is the proposition that a study published in *Nature* is an expert source. The other is the proposition that *Encyclopaedia Britannica* is a reliable source. The automated assistant can put these propositions together into a sequence of argumentation that could be used to prove the conclusion that Wikipedia is reliable, as shown in figure 9.

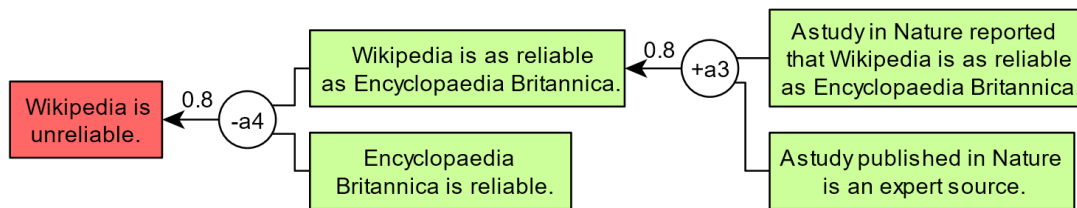


Figure 9: Second Speaker's Argument in the Wikipedia Example

Let us say that the audience knows that the journal Nature is one of the leading scientific journals and that the articles published in it are subject to rigorous peer review. Argument a_3 is an instance of the argumentation scheme for arguments from expert opinion and since the audience has a high regard for the journal Nature, it finds the argument very strong. Accordingly, let us evaluate the strength of this argument at $a_3 = 0.8$. Let us say as well that the audience has an equally high regard for the reliability of Encyclopaedia Britannica is a source and so they accept the proposition that Encyclopaedia Britannica is reliable. Carneades automatically calculates that the proposition that Wikipedia is as reliable as Encyclopaedia Britannica based on argument a_3 . Given these assumptions, Carneades automatically evaluates the second speaker's argument as shown in figure 9. The con argument a_4 defeats the first speaker's ultimate conclusion that Wikipedia is unreliable, because the bottom premise of a_4 is accepted by the audience based on common knowledge, and the top premise of the argument is supported by a strong pro argument that has both premises accepted by the audience.

What the automated argument assistant has done is finding a con argument, namely the second speaker's argument shown in figure 9, to rebut the first speaker's pro argument, the argument shown in figure 8. This counterargument attacks the conclusion of the prior argument. The second

speaker has attacked the original argument of the first speaker, but has not succeeded in refuting the original argument. In fact the outcome is a deadlock. There is a strong pro argument supporting the conclusion that Wikipedia is unreliable and a strong con argument attacking the conclusion that Wikipedia is unreliable. Both arguments are equally strong, and therefore neither side has met its burden of proof according to the preponderance of evidence standard. The problem for both sides is to break the deadlock by finding another argument that does not even have to be all that strong to tilt the burden of proof towards the other side.

9 A legal example

In this section we show how the Carneades system for argument invention can be applied in a more detailed way to a hypothetical legal case concerning software copyright licensing. This example is based on an application developed using an earlier version of Carneades to help software developers analyse software licensing issues.⁶⁶ The main issue in the case is whether a fictional argumentation software, "ArgSys", roughly based on an earlier version of the Carneades software, may use a particular open source software license, the Eclipse Public License (EPL).

Open source software licenses grant developers the right to reuse the software in their own programs, which are then "derivative works" of the licensed software. Some open source licenses, however, called reciprocal licenses, require these derivative works to be licensed using the same open source license,

⁶⁶ Thomas Gordon *Analyzing Open Source License Compatibility Issues with Carneades* Proceedings of the Thirteenth International Conference on Artificial Intelligence and Law (ICAIL-2011: no editor given) (New York: ACM Press, 2011) pp.50-55.

or a compatible open source license. Whether or not a particular use of the licensed software creates a derivative work invoking the reciprocity condition is not always entirely clear. A clear example of such a use is textually modifying the source code of the licensed software. But if the licensed software is used only by linking to it, using it as a software library, the legal opinions diverge. Some lawyers, including those at the Free Software Foundation (FSF), argue that linking to a library does create a derivative work. Other copyright law experts, such as Lawrence Rosen, argue that linking alone is not enough to create a derivative work. This issue has yet to be definitively decided by the courts.

Figure 10 shows the argumentation in this fictional case. On the left, shown in the diamond, is the main issue, i_1 , about whether the ArgSys software may use the EPL or not.

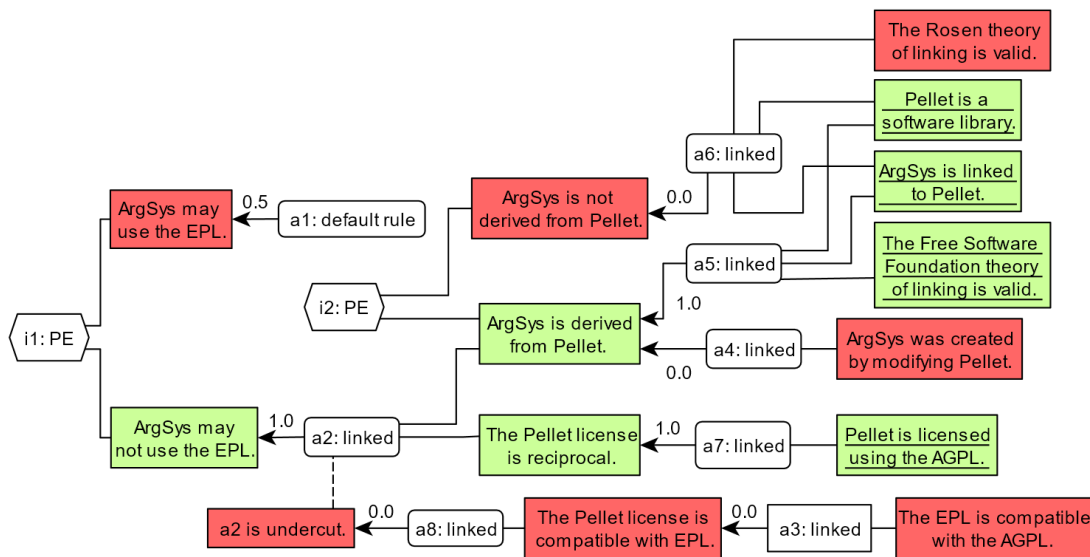


Figure 10: A Carneades Argument Diagram of the Legal Example

Each of these two positions is shown in the boxes just to the right of the issue and each position is supported by an argument, a_1 and a_2 , respectively. The argument for being able to use the EPL, argument a_1 , makes uses of a domain-

specific argumentation scheme, called the “default rule”, with a low weight, 0.5, and no premises. The default rule simply states that a software project may use any open source license it chooses, if there is no stronger argument against using the license. Argument a_2 is, however, such a stronger con argument, where by con argument here we mean an argument for a contrary position, the position that the ArgSys software may not be licensed using the EPL. Argument a_2 states that the EPL may not be used if ArgSys is derived from a software called “Pellet” and Pellet is licensed using a reciprocal license, unless the EPL is compatible with this reciprocal license. The exception for compatible licenses is represented as an undercutting argument, a_8 , of argument a_2 . A subsidiary issue in the case is whether the ArgSys software is derived from Pellet. There are two arguments for this being the case: a_4 , a failed argument stating that ArgSys was developed by modifying Pellet; and a_5 , a successful argument stating that ArgSys is linked to Pellet, as a library, and that it is thus a derivative work, assuming the FSF’s theory is valid that linking creates a derivative work. The assumptions made in the case are underlined in the diagram. Given these assumptions and arguments, the position of issue i_1 that the ArgSys may not use the EPL has the better support, because ArgSys has been linked to Pellet, which is licensed using the reciprocal license, the AGPL, and it has been assumed that the FSF’s theory of linking is valid.

10 Conclusions

It is concluded that Debater works in a way very similar to an argumentation mining system in that it searches a given text for any arguments that can be found in a text. These might be good arguments, or they might be bad ones. In contrast,

Carneades is a normative system that makes use of argumentation schemes.⁶⁷ The system of argument mining described above also utilises argumentation schemes, and to that extent also has a normative component. Debater picks out arguments from a text by polarity, and other requirements,⁶⁸ but when it finds an argument, there is no guarantee that this argument meets normative standards. It just looks for any arguments that are pro or con the designated claim in a text such as Wikipedia.

An illustration of a practical problem of using Debater to identify CDC's concerns the kind of case where a CDC is embedded into a longer Wikipedia sentence. In one of the examples Levy et al. consider,⁶⁹ the CDC claiming that violence in games hardens children to unethical acts is embedded in a longer sentence which states that two named individuals argue that violence in games hardens children to unethical acts, and goes on to call first-person shooter games murder simulators. This text contains several arguments combined in one long sentence where the CDC appears in the middle of the sentence. The task here is finding the boundaries of the text that can be classified as a CDC, so that the non-useful parts of a longer sentence in the natural language text can be excluded.

This suggests to us that Debater goes over the borderline of pure argument mining at some points. The arguments it finds are not the same as the ones put forward by the authors of the texts. Rather, Debater extracts elements of these arguments (the CDC's) which are then reused to construct related, but new arguments. If this is correct, then Debater can be said to "invent" new arguments after all. We conclude that although Debater in the main fits the category of argument mining very well, there are reasons to think it carries out some tasks

⁶⁷ Douglas Walton and Thomas Gordon, "The Carneades Model of Argument Invention" (2012) 20(1) *Pragmatics and Cognition* 1-31.

⁶⁸ Levy et al., *supra* n. 20.

⁶⁹ *Ibid.*, p. 1489.

that fit under the category of argument invention. Here is an attempt to sum up the difference between Debater and Carneades as they relate to *inventio*. Both systems find arguments, but Carneades does this by inventing new arguments using a rule-based inference engine, argumentation schemes and a knowledge base. Debater finds arguments by mining for arguments in full text databases, such as Wikipedia. These arguments can be new, but were not constructed by applying schemes.

This paper has shown how new knowledge-based computational systems in AI provide a critical mass that can help the field of argumentation studies to move forward to achieve progress towards the ancient goal of building argument invention methods to help an arguer with the task of finding arguments to prove a disputed claim. Carneades is a true argument invention tool for several reasons. Carneades is actually an implemented software argument assistant that can help a user construct arguments to prove a designated proposition. Carneades has two knowledge bases that it searches through to find arguments. One is a set of propositions representing the commitments accepted by the audience. The other is a set of argumentation schemes. Debater searches through a natural language database such as Wikipedia where claims have to be dug out of the text. Debater makes extensive use of tools from computational linguistics to extract CDC's from a natural language text of discourse that is used as its database.

However, as was shown in the paper, other knowledge-based AI argumentation systems that employ argumentation tools, such as argument mapping technology, argumentation schemes and different kinds of argument attacks, can also be used to support the task of constructing arguments from a knowledge base. Even so, a qualification needs to be made in this regard. The argumentation tools mentioned in this paper provide some support for argument invention, but not to the same degree. For argument invention, the most important tool is an inference engine that is capable of using a knowledge base

of argumentation schemes to search for arguments. Of the tools discussed in the paper, only the Carneades argument assistant provides this kind of support for argument invention. ASPIC+ and Deflog have been viewed by some as supporting argument invention, whereas in our view they only support argument evaluation, since their knowledge bases are comparable to (static) argument graphs, rather than argument invention in the sense of automatically constructing arguments by instantiating argumentation schemes. Of the three systems, only Carneades supports the argumentation schemes with variables and, in particular, argumentation schemes with second-order variables. Rules in ASPIC+ and Deflog are propositional, without support for any kind of scheme variables.

Acknowledgements

We would like to thank The Social Sciences and Humanities Research Council of Canada Insight Grant 435-2012-0104 and the Fraunhofer Fokus for support of the work in this paper.