Beyond boxes and arrows: argumentation support in terms of the knowledge structure of a legal topic

Maaike Schweers & Bart Verheij
Artificial Intelligence, University of Groningen

Abstract. Today's argumentation software mostly emphasizes the logical structure of reasoning, and especially the structure as it can be represented in boxes-and-arrows style diagrams. In this paper an alternative way of providing argumentation support is proposed. A content-oriented, relatively lowtech tool is presented, based on the knowledge structure of a legal topic, inspired by the structure of legal treatises. The design for this system builds on research in the areas of law, logic, argumentation theory and cognitive ergonomics. A small user evaluation has been performed to examine the usefulness of the system. We show that supporting legal reasoning based on the knowledge structure of a legal topic can be an interesting foundation for argumentation software.

1. Introduction

Recently there is an increase in the attention for software designed for the support of argumentative tasks (Kirschner et al. 2002, Verheij 2005b, Reed & Norman 2003) in the context of law. A recently organized conference on this topic (with special focus on reasoning about evidence) is a case in point.[1] Several tools were presented, with different kinds of focus and designed with different aims. For instance, Van Gelder presented Rationale, a follow up of the Reason!Able package (Gelder 2001), and focusing on generic argumentation and the improvement of reasoning. Gordon showed Carneades (Gordon et al. 2007), in which special attention is paid to different proof standards. Loui, the initiator of the Room 5 system (Loui et al. 1997), raised doubts about the feasibility of the goal of improving reasoning using software tools. Prakken presented work on sense-making software for crime investigation (Bex et al. to appear). Reed discussed the latest developments of the Araucaria system (Reed & Rowe 2004), and its support of different styles of argument diagramming. Verheij discussed the visualization and evaluation of defeasible arguments with warrants, in the context of the ArguMed system (Verheij to appear). Walker emphasized connections between rule-based reasoning and the evaluation of evidence (cf. Walker 2007). The proceedings of the conference will be published in a forthcoming issue of the *Law, Probability and Risk* journal.

[1] The conference 'Graphic and Visual Representations of Evidence and Inference in Legal Settings' was held at the Cardozo Law School in New York, and organized by Peter Tillers, Henry Prakken, Thomas D. Cobb and Jonathan Gottfried. See http://tillers.net/conference.html.

A common trend underlying this style of work on argumentation support software is the graphical representation of arguments, often influenced by the argument diagramming systems by Wigmore and Toulmin (dating from the early and middle of the 20th century, respectively; Wigmore 1931, Toulmin 1958). The graphical representation usually consists of boxes that correspond to the propositional content of an argumentative move ("The accused is the murderer", "Three independent witnesses have seen the accused kill the shop owner", "The accused has a water-tight alibi") and arrows to express relations ("The accused is the murderer since three independent witnesses have seen him kill the shop owner", "The accused is not the murderer since he has a water-tight alibi") (Figure 1; made with ArguMed).

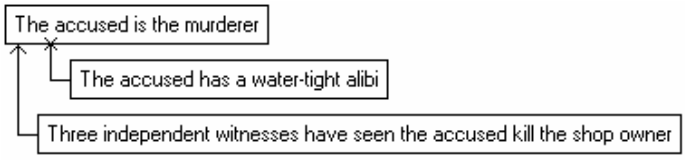

Figure 1: An argument as a boxes-and-arrows diagram

Boxes-and-arrows diagrams of arguments have proven to be useful tools, especially for analytical and educational purposes, but there are many topics that deserve further research attention. Three challenges need especially urgent attention (Verheij to appear; presented at the mentioned New York conference): natural design, usefulness and content. Meeting the challenge of *natural design* does not only involve the exploration and invention of natural ways of representing arguments and constructing them, but also experimental proof that one system is more natural than another. A similar point holds for the challenge of *usefulness*. Not only is it relevant to look for tasks and contexts in which argumentation software can genuinely increase productivity, also is there a need to experimentally show such usefulness. The challenge of *content* is based on the idea that it is not in the first place abstract analytical support that can be of help for professionals working in the domain of law, but instead content-oriented support.

This paper reports on a recent attempt to deal with the challenges mentioned. The research question under investigation was how content can be usefully and naturally integrated in an argumentation support tool. We hypothesized that by integrating the knowledge structure of a legal topic in argumentation software usefulness and naturalness would increase.

Our research method consisted of the systematical development and investigation of a prototypical system. In the pre-design phase of the project, the challenge of usefulness was dealt with by interviews with legal professionals in order to find out what they considered to be useful argumentation support. In this phase, the challenge of naturalness was approached by developing the design requirements following lessons learnt in cognitive ergonomics research. A small post-implementation user study was used to establish whether we were on the right track with respect to the goals of naturalness and usefulness. The challenge of content was approached by our focus through-out the project on a way to integrate argumentation-oriented with content-oriented support. The main contributions of the research are the implemented prototype ArguGuide (section 2) and its accompanying pilot user study

(section 3). As will be seen later (section 4), ArguGuide has a rather different design style from tools based on boxes-and-arrows diagramming.[2]

2. ArguGuide

In this section we discuss the design considerations of the ArguGuide prototype.

2.1. Integrating the knowledge structure of a legal topic in argumentation software

The central design issue that we had to solve was to find a way in which the knowledge structure of a legal topic could be integrated into an argumentation support tool. Initially, we saw two reasonable options, each at one end of a spectrum of possibilities. The first option (inspired by our previous research dedicated to the argumentation support tool ArguMed; Verheij 2003a, 2005b) was to use a boxes-and-arrows diagram of the central logical relations of a legal topic as a starting point. In this option, concrete arguments like the one in Figure 2 (about an unlawful act by Peter; visualized in the ArguMed tool) would be adapted to a generic argument structure that could then form the basis of arguments by the user.

Figure 2: A legal argument in boxes-and-arrows format

The second option was inspired by considering where a legal professional would look for the knowledge structure of a legal topic: in an authoritative treatise on the topic. We considered that already the table of contents of a treatise provides the knowledge structure of the legal topics dealt with in the treatise in a rudimentary form. Perhaps providing the structure of a legal topic in the simple hierarchical format of a treatise table of contents would already be helpful.

Both options, boxes-and-arrows diagramming and a table of contents approach, seemed to have drawbacks from the perspective of our goals. With respect to the first option, we considered that experience with the ArguMed system and its accompanying user studies (Verheij 2005b) had suggested that boxes-and-arrow diagramming of legal arguments would be of limited help to professionals. The second option had as its drawback that providing knowledge structure as a table of contents seemed to be a far cry from genuine support for the drafting of legal arguments.

Considering the drawbacks of the extreme options we looked for a compromise, in an attempt to have the best of both: we took the hierarchical structure of a table of contents and integrated it with a logical relation that is essential for legal

[2] ArguGuide has been developed and implemented by Maaike Schweers during her Master's project. She was advised by Bart Verheij.

professionals: whether an argument is supporting or attacking a claim. Figure 3 shows the result, as it appeared in our ArguGuide prototype.

We used Dutch tort law as an example. We hypothesized that a structure as in Figure 3 contains valuable information for a legal professional for it contains the central elements that could be part of a written argument. In addition these structures have resemblances with the construction of an argument and therefore can help a lawyer to construct one of his own. Also there remain similarities with structures in logic. The hierarchical construction has some resemblance with the construction of a logical proof. For example, to show that unlawfulness is the case, one has to consider the subelements in the structure (violation of a right; violation of a statutory duty; violation of unwritten law; ground for justification). Another resemblance with logic is that in this knowledge structure it is clear what supports the validation of an element and what pleads against it. In ArguGuide, these are coloured green and red respectively; in the rendering here, black bars are used to indicate counterarguments. Also the structure is close to the table of contents structure of legal treatises and therefore familiar for most lawyers.

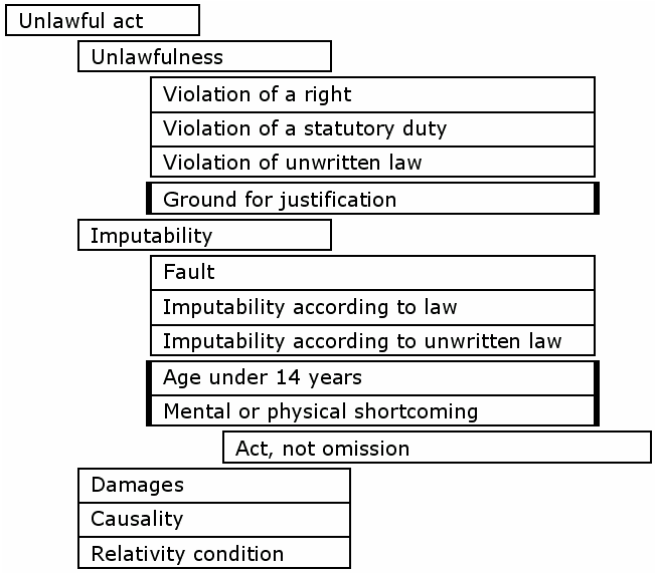

Figure 3: A knowledge structure in the ArguGuide system

2.2. Design requirements

Our design for the ArguGuide prototype is founded on a study of argumentation software, argumentation theory and cognitive ergonomics (Wickens et al. 1998) and is enhanced by interviewing two experts in the practice of law. In conclusion of this study, several design requirements have been defined, of which we here provide a summary.

1. The knowledge structure of a domain must be the foundation for ArguGuide.
2. The legal sources relevant for the legal topic (legal codes and case law) must be contained in the system.
3. The legal sources must be related to the knowledge structure provided.

4. The system should allow the drafting of a plea about a legal case in the form of a written document.
5. The system should allow the drafting of arguments about elements of a case.
6. The system should be usable, independent of the represented side in the legal conflict. The system should also be usable for the decision maker.
7. Still, as became clear from our two expert interviews, the system should provide information concerning burden of proof related knowledge, especially whether defending a particular legal position would help one party or the other.
8. The system should warn whether an element in an argument was overlooked. This "checklist" functionality was also mentioned by one of the experts.
9. The system should be useful for users of different levels of expertise and using different working methods.
10. The system should be consistent with already existing working standards.
11. The system should be user friendly.

Requirements 1-3 pertain especially to the challenge of content mentioned in the introduction, requirements 4-8 about the system's functionality deal with the challenge of usefulness, while requirements 8-11 focus on the challenge of naturalness.

2.3. The resulting design

The screen setup of ArguGuide is shown in Figure 4. The knowledge structure is on the left side (requirement 1). This structure is used as a menu to relevant legal sources (requirements 2 and 3). When the user clicks on an element of the knowledge structure, relevant information about the element opens in the top-right window. This information contains relevant legal articles and other relevant sources, such as case law or text from a legal treatise. On the bottom-right, there is a text-window where the user can formulate an argument about the case he/she is working on (requirement 4). Also the user has the possibility to make notes about a single element of the knowledge structure (requirement 5). This can be done by clicking on the icon next to an element. A little window opens where the user can write a note about this element.

Colour indicates whether an element pleads for or against an element (requirement 7). Still, by the way in which the menu works and by the information that the system contains, it is suitable independent of the side represented (requirement 6). In this connection, it is noteworthy that both experts mentioned that they employ the same methods whether they represent one side or the other because all relevant information has to be examined in order to prevent insufficiencies in the argumentation. Our design supports the checklist functionality of requirement 8, but does not actively give a warning when the user overlooks something in his/her argument. However, the presentation of the knowledge structure stimulates that the user takes all elements of the legal topic into account.

The knowledge menu in ArguGuide is designed to be flexible for different levels of expertise and working methods (requirement 9). For this, the knowledge structure providing access to legal sources, is presented, but users are not forced to make use of it. At first only the top level elements are shown. The user can then click on the elements and if there are subelements considering this element these will be presented. The idea for this design requirement originated from the design of Elenchos (Scaltsas 1998), a system that shows a philosophical argument. Because of this design the user can navigate in an explorative way as well as on a specific level (Wickens et

al. 1998), which is relevant for users with different levels of expertise and working methods. The menu has resemblances with the index of a treatise and therefore supports a natural way of searching for a specific element. The relevant sources (legal codes law and case law) are accessible, but only when a user wants to elaborate upon an element. The user is free to look for further information or to write argumentation at any time. By this freedom, both experienced and inexperienced users can use the system for writing a plea. Experienced users can look quickly into the elements and then construct an argument, while an inexperienced user can first get some more information from the sources.

By the possibilities to freely write an argumentative text and to take notes, the use of the system allows working in a way that is normal today, i.e., by writing at a desk using a word processing package (requirement 10).

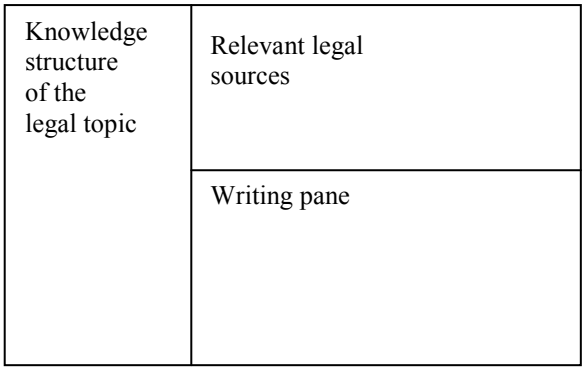

Figure 4: The screen setup of the ArguGuide prototype

The design of ArguGuide makes the system user friendly in various ways (requirement 11). All relevant legal sources that concern a task can be found close to each other (Wickens & Carswell 1995). In ArguGuide, it is easy to switch between the legal codes view and the case law view, by using the dedicated tabs on top of the screen. In the knowledge menu, it is shown which elements contain subelements and can be 'opened' by showing a plus sign next to this element. When subelements are shown there is a minus sign visible, and clicking it closes the subelements. The last element that the user has clicked on, for which the relevant legal sources are shown, is coloured darker than the other elements. In this way, the system is transparent with respect to what is shown on the interface (Norman 1986). The system is compatible with a user's existing mental models (cf. Roske-Hofstrand & Paap 1986) that originate amongst others from experiences with a Windows environment. The knowledge menu can be regarded as a kind of roadmap of the system, hence making the system's search space more transparent (Vicente & Williges 1988).

3. Evaluation (pilot study)

3.1. Setup of the user study

Testing the usefulness of a system requires that effectiveness, efficiency and satisfaction of representative users doing representative tasks are assessed (as concluded by Bevan & Macleod 1994). We have tried to approximate this ideal in a small pilot study. Our study, comprising of eight test subjects, mainly consists of

observing the working methods of the subjects and of their opinions of the system. Direct observation of behavior is recognized as a useful knowledge gathering method, provided it is done in a careful and systematic way and with awareness of for instance the possibility of observer bias (Mook 2001, chapters 4, 5). We have chosen the observational approach as we considered it to be the most appropriate one. For instance, a comparative study was not an option, as there is no system that is sufficiently close in functionality to ArguGuide. The observations are set up to get as much information from the test subjects as possible. First a questionnaire is given to the test subject to inform us about the knowledge and previous experience of the subject. Second the test subject is informed about the setup of the experiment and gets to read a manual of ArguGuide. Then the subject gets to carry out the task for 45 minutes. This amount of time is believed to be sufficient for the subject to form an opinion about the system. Meanwhile the interaction between the subject and the system is observed by using monitoring software. The observant notes which aspects of the system are used in which order and looks for distinct situations in the interaction. After these 45 minutes, the test subject gets a second questionnaire. This questionnaire gets into detail to determine the perceived effectiveness, efficiency and user satisfaction of different aspects of the system. At last the test subject is interviewed by the observant. This interview is meant to be a supplement to the previous questionnaire. Interesting things noted during the observation will be discussed and the test subject is asked if he/she has opinions about the system that did not come up in the questionnaire.

3.2. Test results

The results from the questionnaire show that most test subjects consider the system to be useful. This was also observed. All aspects of the system were used by the test subject. The only exception to this is the functionality to make notes for individual elements of the knowledge structure, which was not used by all test subjects. All subjects were able to carry out the task with the system, even though the working methods differed a lot. Some test subjects started immediately with writing down a draft of the argument, while others started with unravelling the menu and the resources.

The knowledge structure, presented as a menu, was considered to be useful. The questionnaire shows that most test subjects think it is effective as a memento and to prevent the user from coming to a premature conclusion. Also one of the test subjects mentioned in the interview that the knowledge structure encourages the user to work methodically. The colours that are used in the menu and the connection between the structure and the resources are also considered useful. This shows in the results of the questionnaire. During the observations, it became clear that what happens when a menu-item is clicked on is intuitive.

However, the user evaluation also gave rise to doubts about the system. The questionnaire shows that some test subjects did not think the system made it easier to carry out the task. This doubt was explored further in the interviews. Curiously the test subjects who had this doubt were more experienced in the field of law. One possible explanation for this doubt is the rather basic nature of the knowledge that is represented in the ArguGuide prototype. For someone who is already familiar with this knowledge, a system that represents this small amount of knowledge holds no additional value.

Another possible explanation for this doubt is that experts tend to resist towards changing their practices. In this case this not only has to do with inflexibility. In one of the interviews the present practices of lawyers came up. This suggested that

lawyers use their computer mainly as a text processor and in addition as a search engine. They search for case law and other resources on the internet, but in the end they print all their information and phrase their argument based on a desk full of papers. This practice is not possible with a computer system. First of all it is not (yet) possible to see all this information in a clear overview on a computer screen. Secondly, even if you are able to give a clear overview on a computer screen, important information that you get from browsing through a book is lost. For example from the index of a law book you get information about the relation that legal codes have to each other and when you look into a code book you encounter that some codes are more encompassing than others.

4. Discussion

In this section, we address some issues that have arisen while thinking about ArguGuide.

A first issue is how ArguGuide is related to knowledge technology, and in particular to rule-based expert systems, which are the archetypal examples of systems that provide knowledge support. ArguGuide is clearly very different from rule-based expert systems. Especially, it performs no automatic reasoning. The central reason for this deviation is the well-known limited scope of rule-based expert systems in the field of law. Rule-based expert systems are to some extent useful in the field of law, namely in such cases where a legal topic is precisely specifiable in terms of formal rules, but the range is limited (cf. Oskamp & Lauritsen 2002). Our aim was to make a design that is much more widely applicable. The openness of the legal system and the fact that especially - what have been called - hard cases go to trial necessarily leads to the demand of freedom and flexibility by legal professionals. Argumentation software is an attempt to meet these demands (cf. also Gordon 1996). ArguGuide has the same goal, but uses a different approach. Interestingly, knowledge acquisition - in general considered a bottleneck - might take a different shape in a tool like ArguGuide: if our approach turns out to be a useful one, the task of knowledge acquisition is very close to that of writing a legal treatise. Hence it would indeed be a laborious burden, but a feasible one, as the current practice of treatise production shows. In this respect, we expect that the design of good content authoring tools will become an important line of research.

A question that springs to mind is whether this kind of relatively lowtech solution is still Artificial Intelligence. Our answer is that it is not in the sense of performing intelligent autonomous behaviour, but that it is in the sense of containing and exploiting represented knowledge: ArguGuide uses a structured representation (albeit simple; see the issue on logic discussed next), which is used in the working of the system. We are currently considering an adaptation of the technology underlying ArguGuide using Semantic Web standards (Berners-Lee et al. 2001), in particular XML, RDF and OWL (cf. also Verheij 2005a). We expect that this will make it more easy to scale up the system and also to exchange information.

One may wonder why the elements in the knowledge structure of the ArguGuide system are from a logical perspective so loosely linked. It is not even represented whether rule conditions are cumulative or alternative (i.e., logical conjuncts or disjuncts). We did this on purpose since we believe that legal professionals hardly ever need help with the question whether rule conditions are conjunctively or disjunctively related. Note also that the expert treatises that form our inspiration do not

pay much attention to conjunction/disjunction matters. Moreover, extensive experience teaching legal argumentation to law students (Verheij et al. 2004) revealed that apparent conjunction/disjunction errors made by students arose especially from problems with the use of the semi-formal scheme for the analysis of rule conditions that they had to use.

In contrast, there is a need for support with respect to legal content, as evidenced by the market for legal treatises. Similarly, in our teaching experiences, it was obvious that law students had problems with legal content: they do forget which conditions are relevant. As a result, we hypothesized that legal professionals are genuinely supported with hierarchically structured lists of conditions as in ArguGuide. A result of these considerations is that the system contains only a limited amount of logical structuring: relevance (represented by the element-subelement relation) and pro/con (represented by the colouring of an element).

A final point that we want to make is a related, but different development to include content in argumentation software: argumentation schemes (Walton 1996; cf. Verheij 2003b). This is certainly a worthwhile direction of research (e.g., taken up by Reed & Rowe 2004 and Bex *et al.* to appear). Initial experience (also in the ArguMed tool) suggests however that argumentation schemes are especially useful for argument analysis and evaluation, and less for argument drafting, as was our focus.

5. Conclusion

Systems like ArguGuide can be an enhancement in the research considering argumentation support. Boxes-and-arrows-like argumentation software supports structuring and visualizing any argument, while systems like ArguGuide can support formulating an argument considering the content of a specific topic. It remains to be seen however in which kinds of situations professional lawyers will need this style of diagramming. It may turn out that software tools for argument support will prove especially useful by their help to accelerate the virtualization of the legal workplace, including the luxury of full-text search of large amounts of legal codes, court decisions and case files.

More research in the direction of content oriented argumentation software seems interesting. First of all it should be explored how a system like ArguGuide can deal with more content. Especially the scaling up of the number of sources included in the system is required (cf. also the discussion in section 4). It would be interesting to investigate to what extent this can be automated. Secondly it would be good to perform user studies with a larger test group. This could give a significant result about the additional value of a knowledge structure in the support of legal reasoning. The observational study that is performed in this research is only a pilot for such a larger study, but at the same time has proven to give useful insights into possible avenues for the design of content-supporting argumentation tools. Also the resistance that some lawyers have requires serious attention by the field. One step in the right direction could be to study the present argument drafting practices of lawyers.

References

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American*. May 17, 2001

Bevan, N., & Macleod, M. (1994). Usability Measurement in Context. *Behaviour and Information Technology* 13 (132-145).

Bex, F.J., Van den Braak, S.W., Van Oostendorp, H., Prakken, H., Verheij, B., & Vreeswijk, G. (to appear). Sense-Making Software for Crime Investigation: How to Combine Stories and Arguments? *Law, Probability & Risk*.

Gelder, T.J. (2001). The Reason! Project. *The Skeptic* 21 (2), pp. 9-12.

Gordon, T.F. (1996). Computational Dialectics. *Computers as Assistants. A New Generation of Support Systems* (ed. P. Hoschka), pp. 187-203. Mahwah (New Jersey): Lawrence Erlbaum Associates.

Gordon, T.F., Prakken, H., & Walton, D.N. (2007). The Carneades Model of Argument and Burden of Proof. *Artificial Intelligence* 171, pp. 875-896.

Kirschner, P.A., Buckingham Shum, S.J., & Carr, C.S. (2002). *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. London: Springer-Verlag.

Loui, R.P., Norman, J., Altepeter, J., Pinkard, D., Craven, D., Linsday, J., & Foltz, M. (1997). Progress on Room 5. A Testbed for Public Interactive Semi-Formal Legal Argumentation. *The Sixth International Conference on Artificial Intelligence and Law. Proceedings of the Conference*, pp. 207-214. New York (New York): ACM.

Mook, D.G. (2001). *Psychological Research. The Ideas Behind the Methods*. New York (New York): W.W.Norton.

Norman, D.A. (1986). Cognitive Engineering. *User Centered System Design: New Perspectives on Human-Computer Interaction* (eds. D.A. Norman & S.W. Draper), pp. 31-61. Hillsdale (New Jersey): Lawrence Erlbaum Associates.

Oskamp, A., & Lauritsen, M. (2002). Ai in Law Practice? So Far, Not Much. *Artificial Intelligence and Law* 10, pp. 227–236.

Reed, C., & Norman, T.J. (eds.). (2003). *Argumentation Machines. New Frontiers in Argument and Computation*. Dordrecht: Kluwer Academic Publishers.

Reed, C., & Rowe, G. (2004). Araucaria: Software for Argument Analysis, Diagramming and Representation. *International Journal of AI Tools* 13 (4), pp. 961-980.

Roske-Hofstrand, R.J., & Paap, K.R. (1986). Cognitive Networks as Guide to Menu Organization: An Application in the Automated Cockpit. *Ergonomics* 29, pp. 1301-1311.

Scaltsas, T. (1998). Representation of Philosophical Argument. *The Digital Phoenix: How Computers Are Changing Philosophy* (eds. T. Bynum & J. Moor). Oxford: Blackwell Publishing.

Toulmin, S.E. (1958). *The Uses of Argument*. Cambridge: Cambridge University Press.

Verheij, B. (2003a). Artificial Argument Assistants for Defeasible Argumentation. *Artificial Intelligence* 150 (1-2), pp. 291-324.

Verheij, B. (2003b). Dialectical Argumentation with Argumentation Schemes: An Approach to Legal Logic. *Artificial Intelligence and Law* 11 (1-2), pp. 167–195.

Verheij, B. (2005a). An Argumentation Core Ontology as the Centerpiece of a Myriad of Argumentation Formats. *Input Agentlink Argumentation Interchange Format Technical Forum 2005*. Available at http://www.x-opennet.org/aif/.

Verheij, B. (2005b). *Virtual Arguments. On the Design of Argument Assistants for Lawyers and Other Arguers*. The Hague: TMC Asser Press.

Verheij, B. (to appear). Argumentation Support Software: Boxes-and-Arrows and Beyond. *Law, Probability & Risk*.

Verheij, B., Hage, J.C., van der Meer, T., & Span, G. (2004). *Vaardig Met Recht. Over Casus Oplossen En Andere Juridische Vaardigheden (Skilful in the Law. On Case Solving and Other Legal Skills)*. The Hague: Boom Juridische Uitgevers.

Vicente, K.J., & Williges, R.C. (1988). Accommodating Individual Differences in Searching a Hierarchical File System. *International Journal of Man-Machine Studies* 29 (647-688).

Walker, V.R. (2007). A Default-Logic Paradigm for Legal Fact-Finding. *Jurimetrics* 47 (2), pp. 193-244.

Walton, D.N. (1996). *Argument Schemes for Presumptive Reasoning*. Mahwah (New Jersey): Lawrence Erlbaum Associates.

Wickens, C.D., & Carswell, C.M. (1995). The Proximity Principle: Its Psychological Foundation and Its Relevance to Display Design. *Human Factors* 37 (3), pp. 473-494.

Wickens, C.D., Gordon, S.E., & Liu, Y. (1998). *An Introduction to Human Factors Engineering*. New York (New York): Longman.

Wigmore, J.H. (1931). *The Principles of Judicial Proof or the Process of Proof as Given by Logic, Psychology, and General Experience, and Illustrated in Judicial Trials, 2nd Edition*. Boston (Massachusetts): Little, Brown and Company.