How much logical structure is helpful in content-based argumentation software for legal case solving?

Stijn Colen, Fokie Cnossen, Bart Verheij
Artificial Intelligence, University of Groningen

ABSTRACT

Current argumentation support software often employs graphical representations of logical relationships. Little is known about the extent to which logical structuring helps to increase a user's task performance. In this paper, various levels of graphical representation of the logical structure of legal subject matter are experimentally compared in terms of performance. It is shown that logical structuring significantly increases task performance, but we have found no evidence that the extensive representation of logical structure as employed by several contemporary software applications is more effective or usable than a simplified graphical representation that was previously implemented in an application called ArguGuide.

Keywords

Legal argumentation, argumentation support software, legal decision support systems

1. ArguGuide: content-based argumentation software for lawyers

Although research into the topic of knowledge technology for the support of legal professionals started in the 1970s (Rissland et al. 2003), Oskamp & Lauritsen (2002) document that the adoption of AI technology in legal practice has been slow, a situation that these authors attribute to the wide gap between AI researchers and legal practitioners. Currently argument visualization tools are in the focus of attention (cf. Kirschner et al. 2002, Verheij 2005). According to Van den Braak and colleagues (2006), however, research on argument visualization tools is still largely lacking empirical evidence for their benefits as support tools (but see Schank & Ranney 1995, Carr 2003, Pinkwart et al. 2008, van den Braak et al. 2008). The present paper aims at establishing empirical evidence for the usefulness of the design of an argumentation tool for a realistic legal task.

Verheij (2007) has suggested that legal professionals may benefit more by software that provides checklists for legal content (as a kind of 'memory extensions') than by visualizations showing the logical structure of an argument. With this suggestion in mind, the ArguGuide system (Schweers & Verheij 2007, Verheij 2007) was designed to support the task of establishing the legal consequences of a given case situation. ArguGuide is a prototype of an integrated software environment for the preparation of legal argumentative texts, such as a plea note or court decision, developed in collaboration with two legal professionals. There is a writing pane and a pane showing the logical structure of the legal topic. By clicking elements of this 'content map' relevant sources are activated in a pane showing legislation and case law. By its focus on supporting access to legal content more than on argument diagramming, ArguGuide was designed as a mild challenge to 'boxes-and-arrows' software. Recently, an updated, RDF-based version of this system has been implemented.

The logical structuring in ArguGuide's content map is limited to a hierarchical relation of relevant elements of the legal topic, and an indication of the direction of relevance (whether pro or con).[1] In ArguGuide, there is no way to visualize conjunctive or disjunctive relationships between elements, as is often possible in argument visualization packages. This was a deliberate choice, as it was silently assumed that this was the right level of beneficial logical structuring for the task of case solving.

The present study was set up to put this assumption to the test and possibly empirically underpin the design of ArguGuide. To test whether the level of detail in the logical structure of ArguGuide is too low, sufficient, or too high, we experimentally measured case solving task performance in participants who used the hierarchical representations as used by ArguGuide with that of participants who used representations with either more or less graphical elements.

2. Experiment: materials

To test experimentally to what extent logical structuring increases task performance on logical reasoning, participants in our experiment were presented with nine legal cases. Each case consisted of a description of relevant facts and a question to be answered. The participants were given a textual summary of the legal topic. This text contained sufficient information to solve the case correctly.

Because our participants were laypersons, a balance had to be struck between case complexity and legal realism. Four fields of Dutch civil law were selected: product accountability ('productaansprakelijkheid'), tort law ('onrechtmatige daad'), breach of contract ('tekortkoming in nakoming') and expiration ('verjaring'). Care was taken not to overcomplicate the legal structure of the schemes, to minimize the possibility of reasonable differences in interpretation of the legal subject, which could have distorted our results. For example, in Dutch tort law, an unlawful act requires unlawfulness, which can consist of the violation of a right, statutory duty or unwritten law. However, the distinction between these three is often vague, even for legal professionals; hence several reasonable interpretations can occur side by side. Explaining the intricacies of these legal terms is difficult and distracts from the main legal reasoning task performed by participants. Therefore, the legal scheme was simplified to avoid the issue. In this specific case, the legal element 'ground for unlawfulness' was introduced to cover the three possible kinds of violations, thereby maintaining the structure of the legal field but simplifying it in order to prevent confusion.

[1] Cf. Cato's factor hierarchy (Aleven & Ashley 1997, Ashley & Aleven 1997).

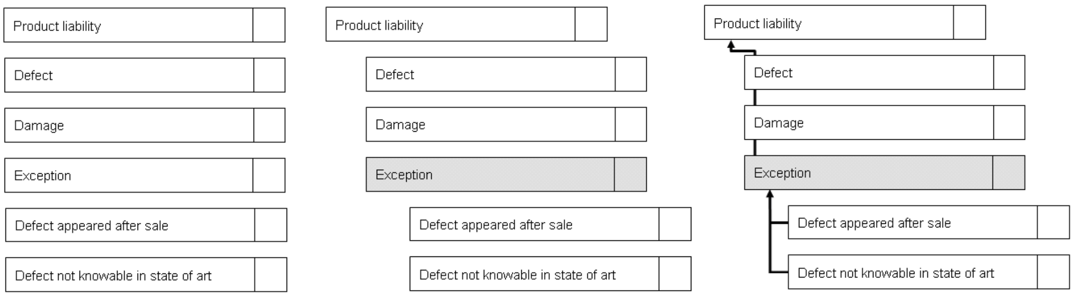
Product liability
Defect
Damage
Exception
Defect appeared after sale
Defect not knowable in state of art
Product liability
Defect
Damage
Exception
Defect appeared after sale
Defect not knowable in state of art
Product liability
Defect
Damage
Exception
Defect appeared after sale
Defect not knowable in state of art

Figure 1: The three experimental conditions: flat, hierarchical and augmented-hierarchical

The logical structure of the legal topic was presented in a diagram, as a graphical aid illustrating the explanatory text. Three different diagram types were used, expressing different levels of logical structuring (Figure 1).

On the left, the *flat condition* is shown. Here the factors relevant for solving the legal case are simply listed. In the center, an example of the *hierarchical condition* is presented. The factors are ordered hierarchically, expressing relevance. For instance, the diagram shows that product liability depends on the occurrence of a defect, damage and exceptional circumstances. The latter is negatively relevant, which is indicated by the use of a shaded background color. The hierarchical condition corresponds to the original version of ArguGuide (Schweers & Verheij 2007). The third condition, the *augmented-hierarchical condition*, is shown on the right. It shows the hierarchical relation of relevance and the direction of relevance as in the hierarchical condition, but in addition shows the logical relation between relevant factors: disjunctive or conjunctive. In the example, product liability occurs when there is a defect *and* damage *and* there is no exception. Such conjunction of factors is indicated by an arrow going through the factors. An exception occurs when the defect appeared after sale *or* was not knowable in the technical state of the art. This kind of disjunction of factors is represented by an arrow allowing parallel paths. The diagram format of this condition is a kind of flow-chart, capitalizing on the mental model of a stream working its way towards its source or root node. In an SQL database query task, a related kind of diagram improved production and comprehension performance (Young & Shneiderman 1993).

The task of case solving consisted of two parts. First, the participant had to fill out the boxes alongside a factor, indicating whether the factor held for the case (writing a check mark, √), whether it failed (writing a cross, ×), or whether the factor was not relevant for solving the case (leaving the box blank). The topmost box in the diagram, representing the legal issue at hand, had to contain a check mark or a cross, thereby providing an answer to the legal question posed. Second, the participant was asked to write down the case solution in a few sentences. These written case solutions were not used for performance measurement, but were meant to ensure that participants put sufficient effort into the case solving process by requiring them to verbally explain their answer.

The experiment was preceded by an explanatory text with examples of case solving. Using these examples, the meanings of indentation, arrows and negation in the diagrams (if applicable to the experimental condition) were explained. Also, a sample case was solved in a step-by-step manner. The experiment was set up to take approximately 60 minutes, but no time restrictions were imposed upon the participants.

The experiment was conducted with pen and paper, not within a software environment. In this way, no side effects of the particular software implementation (such as a participant's experience with an operating system or user interface glitches) would influence the experiment's outcome. For the experiment's goal, namely testing task performance effects of different graphical domain representations, this choice has no effect.

3. Experiment: design

In total, 44 participants, mostly students, took part in the experiment, 30 men and 14 women. One participant was excluded due to non-conformity to the instructions. Because we wanted to control for the effects of formal logical training, we recruited 21 students from Artificial Intelligence with a curriculum containing extensive logical training, 14 students from Psychology, who normally do not take courses on logic, and 10 further participants, enrolled in another study or not enrolled, as controls. Law school students were not eligible to take part in the experiment to exclude the effect their degree of domain knowledge might have had on their task performance. All psychology and most artificial intelligence students taking part in the experiment did so in exchange for obligatory course credits.

We used a between-subject design, where participants were quasi-randomly assigned to the three experimental conditions (Flat, Hierarchical, Augmented), controlling for type of study (Artificial Intelligence, Psychology, Other) (see Table 1). We chose not to use a within-subject design, because we wanted to test a number of different logical structures and legal constructs, and also vary the difficulty of the legal cases; controlling for these factors would have put too many restrictions on the construction of the legal cases.

3.1 Performance measures

Performance was scored on six different logical dimensions: errors in (1) legal correctness; (2) logical correctness; (3)

Table 1: Participant division between experimental conditions

		Type of study		
		AI	*Psychol-ogy*	*Other*
Condition	*Flat*	7	5	3
	Hierarchical	7	4	3
	Augmented	7	4	3

completeness; (4) efficiency; (5) distinguishing conjunctive/disjunctive relationships; and (6) recognition of negation. The errors counted have been listed in Table 2.

Legal correctness errors are made when participants derive invalid conclusions from case texts. For example, a participant may conclude that someone cannot be held accountable for his or her actions, even though accountability legally holds true. Legal correctness is measured per case by counting the number of factors and conclusions marked incorrectly.

Logical correctness errors occur when a logically wrong conclusion is drawn, for instance, when the conclusion of a rule with conjunctively connected antecedent is drawn while an element of the rule's antecedent is false. A 'closed world assumption' is used: the given factors are assumed to represent all existing reasons for a conclusion (hence our use of the biconditional operator ↔ in Table 2). Without this assumption, some errors could be attributed to external, missing reasons.

Table 2: Possible errors measured

Description	Formal notation
Legal correctness	
Incorrect marking of a factor or conclusion	
Logical correctness	
Invalid derivation using a conjunctive rule when at least one conjunctive element is false	$\frac{a \wedge b \leftrightarrow c,\ \neg a}{c}$
Invalid derivation using a conjunctive rule when all conjunctive elements are true	$\frac{a \wedge b \leftrightarrow c,\ a,\ b}{\neg c}$
Invalid derivation using a disjunctive rule when all disjunctive elements are false	$\frac{a \vee b \leftrightarrow c,\ \neg a,\ \neg b}{c}$
Invalid derivation using a disjunctive rule when at least one disjunctive element is true	$\frac{a \vee b \leftrightarrow c,\ a}{\neg c}$
Completeness	
A conjunctive element needed for a derivation is not verified while another conjunctive element holds	$a \wedge b \leftrightarrow c,\ a$ Participant does not verify b
A disjunctive element needed for a derivation is not verified while another disjunctive element fails	$a \vee b \leftrightarrow c,\ \neg a$ Participant does not verify *b*
Efficiency	
A conjunctive element is verified while another conjunctive element fails	$a \wedge b \leftrightarrow c,\ \neg a$ Participant verifies *b*
A disjunctive element is verified while another disjunctive element holds	$a \vee b \leftrightarrow c,\ a$ Participant verifies *b*
Distinguishing conjunctive/disjunctive relationships	
Invalid derivation using a conjunctive rule, drawing a conclusion as if it were disjunctive	$\frac{a \wedge b \leftrightarrow c,\ \neg a,\ b}{c}$
Invalid derivation using a disjunctive rule, drawing a conclusion as if it were conjunctive	$\frac{a \vee b \leftrightarrow c,\ \neg a,\ b}{\neg c}$
Recognition of negation	
Invalid derivation if a negated element holds	$\frac{\neg a \wedge b \leftrightarrow c,\ \neg a,\ b}{\neg c}$
Invalid derivation if a negated antecedent fails	$\frac{\neg a \wedge b \leftrightarrow c,\ a,\ b}{c}$

A participant makes a *completeness* error when an element that can complete a derivation is not verified.

An *efficiency* error occurs when a participant checks a factor of a derivation that is already known to fail. For this dimension, the difference between the optimal number of checked factors and the actual participant's number of checked factors was measured. A problem with this way of measuring efficiency errors is that it does not take into account in which order a participant checks factors. It is possible that checking a factor turns out to be redundant only after a further factor is checked. In our setting there is no obvious way to deal with this issue, as our pen-and-paper approach does not show the order in which factors are checked.

An error concerning the *distinction of conjunctive/disjunctive relationships* occurs when a conclusion is drawn that can only be explained by assuming that the wrong logical relation between factors is used.

A *recognition of negation* error occurs when a derivation can only be explained by assuming that a negatively relevant factor was taken as being positively relevant.

3.2 Data analysis

Since the performance measures were counts of errors made by participants, the Poisson distribution was used for statistical testing. Normal distribution of the data was not expected, and Poisson loglinear models were used to fit the data. We used the Wald Chi-Square test to test for significant effects of experimental condition or type of study. When this multivariate analysis showed a significant effect, univariate pairwise comparisons were used to determine the univariate effects of the variables.

4. Experiment: results

Table 3 shows the average number of errors per condition, while Table 4 shows the averages per type of study.

4.1 Legal correctness

The Poisson loglinear model showed that the experimental condition had a significant effect on the number of legal errors (χ^2(2, N=43)=12.23, *p*<.01). Pairwise comparisons indicated no significant differences between the hierarchical and augmented-hierarchical conditions, but participants working with the hierarchical condition made significantly fewer legal errors than those working with the flat condition (*p*<.01), as did those working with the augmented-hierarchical condition compared to the flat condition (*p*<.01). The model also showed that type of study did not significantly influence legal correctness performance.

4.2 Logical correctness

Analysis of the logical correctness data showed that experimental condition significantly influenced performance (χ^2(2, N=43)=13.61, *p*<.01). No significant differences were found between the hierarchical and augmented-hierarchical condition, but participants in the hierarchical condition made significantly fewer logical errors than those in the flat condition (*p*<.05) while participants in the augmented-hierarchical condition also performed significantly better than those in the flat condition (*p*<.01).

Additionally, type of study was found to be a significant factor in logical correctness performance (χ^2(2, N=43)=22.74, *p*<.001). Students enrolled in Artificial Intelligence performed significantly better than Psychology students (*p*<.001), as well as significantly

Table 3: Average number of errors per condition (and standard deviation)

	Flat	*Hierarchical*	*Augmented*	*Overall*
Legal correctness	5.47 (3.94)	3.29 (2.67)	3.07 (2.43)	3.98 (3.23)
Logical correctness	3.00 (2.36)	1.50 (2.07)	1.14 (0.77)	1.91 (2.01)
Completeness	4.87 (5.89)	2.29 (3.15)	2.00 (2.00)	3.09 (4.20)
Efficiency	2.82 (1.47)	2.61 (1.28)	2.87 (1.31)	2.77 (1.33)
Conjunctive/disjunctive	1.93 (1.91)	0.93 (1.21)	0.79 (0.89)	1.23 (1.48)
Negation	Not analyzed (see text and Table 5)			

Table 4: Average number of errors per type of study (and standard deviation)

	AI	*Psychology*	*Other*
Legal correctness	3.24 (2.10)	4.69 (4.19)	4.67 (3.84)
Logical correctness	0.90 (1.09)	2.38 (1.98)	3.56 (2.51)
Completeness	2.33 (3.76)	3.08 (2.66)	4.89 (6.43)
Efficiency	2.80 (1.00)	2.46 (1.32)	3.19 (2.02)
Conjunctive/disjunctive	0.62 (0.74)	1.92 (1.89)	1.67 (1.66)
Negation	Not analyzed (see text and Table 5)		

better than participants in the Other category (p<.01). No significant differences between Psychology students and participants in the Other category were found.

4.3 Completeness

The experimental condition significantly influenced performance on the completeness dimension (χ^2(2, N=43)=22.33, p<.001). No significant effect was found between the hierarchical and augmented-hierarchical conditions, but participants assigned to the flat condition made significantly more completeness errors than those assigned to hierarchical condition (p<.001), and than those assigned to the augmented-hierarchical condition (p<.001).

Type of study had a significant effect on performance (χ^2(2, N=43)=14.82, p<.01). It turned out that participants not enrolled in Artificial Intelligence or Psychology significantly underperformed compared to students in the aforementioned groups (p<0.01 and p<0.05, respectively). No significant differences were found between students in Artificial Intelligence and Psychology.

4.4 Efficiency

No significant influence on the efficiency dimension was found.

4.5 Distinguishing conjunctive/disjunctive relationships

The experimental condition significantly influenced performance in the distinguishing conjunctive/disjunctive relationships dimension (χ^2(2, N=43)=8.21, p<.05). There was no significant difference between performance of participants assigned to the hierarchical and augmented-hierarchical conditions. However, hierarchical and augmented-hierarchical condition participants performed significantly better than flat condition participants (p<.05, and p<.01, respectively).

Additionally, type of study had a significant effect on performance on distinguishing conjunctive/disjunctive relationships (χ^2(2, N=43)=11.42, p<.01). Pairwise comparisons indicated that Artificial Intelligence students outperformed both Psychology students (p<.01) and Other participants (p<.05). No significant differences were found between Psychology students and Other participants.

4.6 Recognition of negation

Most participants did not encounter any measurable difficulty in the recognition of negation. 37 participants made no errors at all, four participants made one such error, while one participant made two errors and another participant made three negation errors. Because of the low occurrence of this type of error, no further statistical analysis was performed. Table 5 shows the number of errors per participant across the three experimental conditions.

5. Discussion

Our main goal was to establish empirical evidence for the design of ArguGuide (Schweers & Verheij 2007, Verheij 2007). For that purpose, we investigated performance differences between three experimental conditions, in order to test how much logical structure in the representation of a legal topic is useful to support a case solving task. Our three conditions (flat, hierarchical and augmented-hierarchical) represent increasing levels of logical structuring. Since the hierarchical condition corresponds to the logical structuring used in ArguGuide, our experiment can test whether the relatively 'low-logic' approach of ArguGuide can be supported by evidence. If it is true that certain logical relations (in particular the conjunctive or disjunctive relation of the elements making up a reason) do not lead to problems when solving legal cases (cf. Verheij 2007), it was expected that we would not find significant performance differences between the hierarchical and augmented-hierarchical condition. We did expect a significant increase in performance from the flat condition to the hierarchical condition and augmented-hierarchical condition. We will discuss our six dimensions consecutively.

With respect to *logical correctness* we found what was expected: the flat condition gave significantly weaker performance than both the hierarchical and the augmented-hierarchical condition, while no significant difference between the hierarchical and the augmented-hierarchical condition was found. This suggests that the ArguGuide proposal provides 'just enough' logical structure to support the task of legal case solving. Our evidence corroborates the suggestions made by two legal professionals in interviews preceding the ArguGuide design. An explanation for the finding that the augmented-hierarchical condition does not show an increase in performance can be that the additional information with respect to the hierarchical condition, is either obvious after reading the textual information about a legal topic, or can be derived on the basis of a participant's world knowledge.

The results for *legal correctness* were in line with those for logical correctness. Again, the flat condition gave significantly weaker performance than the other two, which could not be distinguished. Since our way of measuring legal correctness is a content-based criterion instead of a criterion based on formal logic, this result can be interpreted as showing that logical structure and legal content are connected in a case solving task: by giving support on logical structure, performance on a content-based criterion is successfully supported. Again, ArguGuide's level of logical structuring was found to give just the right level of support.

Our findings for the *completeness* dimension corroborated what we saw with respect to legal and logical correctness. No difference was found between the hierarchical and the augmented-hierarchical condition, while the flat condition gave significantly weaker performance. The dimension of *distinction of conjunctive/disjunctive relationships* showed the same pattern.

Table 5: Number of participants making errors in the recognition of negation

		Number of errors per participant			
		0	*1*	*2*	*3*
Condition	*Flat*	12	2	0	1
	Hierarchical	12	1	1	0
	Augmented	13	1	0	0

Recognition of negation gave too low error counts to allow reasonable statistics. Presumably, the recognition of negation is a relatively unproblematic part of case solving.

The *efficiency* dimension showed no significant differences between our three conditions. This is surprising, as efficiency is directly influenced by the conjunctive or disjunctive relation between the elements of a reason. The augmented-hierarchical condition gives the best clues for optimal performance with respect to efficiency, so should come out best. Our current explanation for the lack of an effect here is that our experimental method was not sufficiently distinguishing. We already mentioned that we did not take into account in which order the factors were checked, which would have given additional insight into the participants' strategy with respect to efficiency. Possibly an effect of the level of logical structuring can be found when order is taken into account. It is also possible that in our experiment participants were not sufficiently cued to optimize efficiency as they received no specific instruction on this.

We expected performance differences between the three types of study (Artificial Intelligence, Psychology, Other), especially, since Artificial Intelligence students are exposed to a considerable amount of training in logic and might also have strong other skills related to formal structure, such as computer programming. We found that, in logical correctness, Artificial Intelligence students outperformed participants in the Psychology and Other groups. In the completeness dimension, participants in the Other group performed better than Artificial Intelligence and Psychology students. Lastly, Artificial Intelligence students made significantly fewer errors in distinguishing conjunctive/disjunctive relationships than participants in the other two study groups. The only overall trend that can be identified is that Psychology students made significantly more errors than either of the other groups.

6. Conclusion

We have provided evidence that some logical structuring of the relevant legal topic is helpful in a case solving task, but up to a limit. In this way, we were able to empirically underpin a design proposed before (ArguGuide by Schweers & Verheij 2007, Verheij 2007).

We found statistically significant performance differences between the hierarchical and the augmented-hierarchical condition on the one hand and the flat condition on the other. Hierarchical and augmented-hierarchical condition participants outperformed flat condition participants in legal correctness, logical correctness, completeness, and distinguishing conjunctive/disjunctive relationships. This gives reason to believe that the hierarchical and the augmented-hierarchical condition give performance support that is superior to the flat condition.

Nothing in the results of the experiment indicates that statistically significant performance differences exist between the hierarchical and augmented-hierarchical conditions. Since statistically significant differences have been shown to exist between the flat and hierarchical/augmented conditions, this does not seem to be a result of a lack of statistical power. Apparently, the addition of explicit conjunctive/disjunctive relationships to an existing hierarchical structure does not increase performance in any of the experimental assessment dimensions, unlike the introduction of hierarchy and explicit negation such as is the case between the flat and hierarchical condition. This proves the assumption underlying ArguGuide (Schweers & Verheij 2007, Verheij 2007) that in the legal domain, hierarchy and negation offer just enough logical structure for the support of performance in a task of legal case solving. On the basis of our findings, we conclude that graphically showing conjunction/disjunction is redundant since users already extract sufficient logical cues from the meaning of the legal elements themselves.

Acknowledgements

The authors would like to thank Leendert van Maanen, Jelmer Borst and Hedderik van Rijn for discussions on the subject matter of this paper.

References

[1] Aleven, V., & Ashley, K.D. (1997). Evaluating a Learning Environment for Case-Based Argumentation Skills. *The Sixth International Conference on Artificial Intelligence and Law. Proceedings of the Conference*, 170-179. New York (New York): ACM.

[2] Ashley, K.D., & Aleven, V. (1997). Reasoning Symbolically About Partially Matched Cases. *Ijcai 1997. Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence* (ed. M. Pollack), 335-341. Morgan Kaufmann.

[3] Carr, C.S. (2003). Using Computer Supported Argument Visualization to Teach Legal Argumentation. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making* (eds. P.A. Kirschner, S.J. Buckingham Shum & C.S. Carr), 75-96. London: Springer-Verlag.

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

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

[6] Pinkwart, N., Lynch, C., Ashley, K., & Aleven, V. (2008). Re-Evaluating Largo in the Classroom: Are Diagrams Better Than Text for Teaching Argumentation Skills. *Proceedings of the 9th International Conference on Intelligent Tutoring Systems (Its 2008)* (eds. E. Aimeur & B. Woolf), 90-100. Berlin: Springer.

[7] Rissland, E.L., Ashley, K.D., & Loui, R.P. (2003). Ai and Law: A Fruitful Synergy. *Artificial Intelligence* 150 (1-2), 1-15.

[8] Schank, P., & Ranney, M. (1995). Improved Reasoning with Convince Me. *Conference on Human Factors in Computing Systems (Chi '95)*, 276-277. New York (New York): ACM.

[9] Schweers, M., & Verheij, B. (2007). Beyond Boxes and Arrows: Argumentation Support in Terms of the Knowledge Structure of a Legal Topic. *Legal Knowledge and Information Systems. Jurix 2007: The Twentieth Annual Conference* (eds. A.R. Lodder & L. Mommers), 109-118. Amsterdam: IOS Press.

[10] Van den Braak, S.W., Van Oostendorp, H., Prakken, H., & Vreeswijk, G.A.W. (2006). A Critical Review of Argument Visualization Tools: Do Users Become Better Reasoners? *Workshop Notes of the Ecai-06 Workshop on Computational*

Models of Natural Argument (Cmna-06) (eds. F. Grasso, R. Kibble & C. Reed), 67-75.

[11] van den Braak, S.W., van Oostendorp, H., Prakken, H., & Vreeswijk, G.A.W. (2008). Representing Narrative and Testimonial Knowledge in Sense-Making Software for Crime Analysis. *Legal Knowledge and Information Systems. Jurix 2008: The Twenty-First Annual Conference* (eds. E. Francesconi, G. Sartor & D. Tiscornia), 160-169. Amsterdam: IOS Press.

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

[13] Verheij, B. (2007). Argumentation Support Software: Boxes-and-Arrows and Beyond. *Law, Probability & Risk* 6, 187-208.

[14] Young, D., & Shneiderman, B. (1993). A Graphical Filter/Flow Representation of Boolean Queries: A Prototype Implementation and Evaluation. *Journal of the American Society for Information Science* 44 (6), 327-339.