What Information Retrieval Can Learn from Case-Based Reasoning

Marie-Francine Moens
Interdisciplinary Centre for Law and IT
Katholieke Universiteit Leuven, Belgium
marie-france.moens@law.kuleuven.ac.be

Abstract. This article gives an overview of the problems of information retrieval systems that search court decisions. Several solutions and new research directions are suggested. The solutions are inspired by the technologies of current case-based reasoning systems.

1 Introduction

Information retrieval (IR) concerns the retrieval of documents or information from a database of documents that satisfy a user's query. Present-day retrieval systems commonly allow users to express their query with a set of key terms, which are possibly combined with Boolean operators. The result of the search is a list of documents. These are usually sorted by relevance, which most of the time is simply computed as a function of the frequency of occurrence of the search terms in the documents.

Searching for just the right information in legal databases is not always easy. Even database services offered by large providers such as LexisNexis and WESTLAW are not very convenient to use. This is because the systems on which the services run allow little more than full text retrieval augmented by keyword search on manually added subject labels. Essentially, there is no easily discoverable relationship between the text actually used and the concepts it is searched for. The decisions are written in natural language. Natural language is immensely varied. A search query and a subject in a text can be treated and expressed in a large number of ways depending upon the searcher or writer and upon his context. The person who searches usually has little knowledge of the content of the database and how its documents are indexed. Basically, the searcher has to guess the words and phrases that occur in the documents or that are assigned to them. When a large quantity of information is involved, subject labels – besides the huge cost of assigning them manually - are either found to be too general to be useful or too precise to be remembered by users. In addition, it is well known that searching large databases – either by a full text search or by selecting subject labels - results in a large number of potentially relevant documents, which are impossible to consult.

Undoubtedly, there is a need for better retrieval models to be implemented in search engines of legal information systems. The aim of this article is to be an incentive for research into these retrieval models. This article starts with introducing some concepts. Then, we outline what binds case-based reasoning and information retrieval of court decisions with regard to their common goal in information seeking, the case representations, similarity computations and analogical inference. Finally, we summarize possible improvements for building retrieval systems of court decisions.

Marie-Francine Moens, 'What Information Retrieval Can Learn from Case-Based Reasoning' in T.J.M. Bench-Capon, A. Daskalopulu and R.G.F. Winkels (eds.), *Legal Knowledge and Information Systems. Jurix 2002: The Fifteenth Annual Conference*. Amsterdam: IOS Press, 2002, pp. 83-91.

2 Court Decisions

Decisions of judges are a source of law. This is especially the case in countries with a *common law* tradition, but countries with a *civil law* tradition, increasingly value the decisions. Lawyers and courts use past cases in precedent reasoning. They build arguments for their current case based on the prior cases that will support claims of plaintiffs and defendants. The binding effect of legal precedents on subsequent cases is embodied in the doctrine of *stare decisis*, which holds that when a point of law has been settled by a judicial decision, it is not ordinarily to be departed from afterward. The propositions set forth in a precedent that are binding as to subsequent cases are termed the *ratio decidendi* of the precedent.

Perhaps the most important part of a legal decision is the *opinion.* After all, it is in this part that the judge reasons about the case, explains concepts, or interprets legislation. A unique proposition of the law necessary to a decision can seldom be determined. Instead a gradation of propositions ranging in abstraction from the specific facts of the case to more abstract factors and legal issues is found. The facts of a case are very important [15, pg. 31]. The facts are the basis on which the judge has made his conclusion. Factors refer to domain-specific knowledge or stereotypical collections of facts, which tend normally to strengthen or weaken a conclusion that a side should win a particular legal claim [2, 3]. Factors can be related to more abstract legal issues or to legal theories that are employed by courts in deciding claims [26]. Issues are legal questions that must be answered one way or the other in deciding the case [17].

The record of the opinion is stored as *natural language text*. Language through a finite set of words and syntactic constructs offers, however, an enormous potential of combinatorial expression possibilities. For instance, the meaning of a single word in a text is defined in terms of other words. Legal texts use language with all of the advantages and disadvantages of this flexibility. In addition, the law uses the common paradigms of words as found in ordinary language in combination with specific legal interpretations of terms.

3 Case-Based Reasoning with Court Decisions

Case-based reasoning (CBR) is generally concerned with remembering old problem situations and their solution and using these to find a solution, classification or other inference for the current problem. Humans use this form of analogical reasoning in many situations [18, pg. xiv].

A system for automated case-based reasoning requires that:

- Past situations of cases and their solutions or classifications are stored as case representations and indexed in a way that they can be easily searched.
- New problems are compared with past, potentially relevant problems in order to find the most similar ones.
- The solution or classification of the selected previous problem is adopted for the new problem, while possibly requiring some transformations of the current situation.

In the legal field, CBR systems especially aim at inferring arguments for a current case from older cases that are similar to the current case. Important CBR systems among others are TAXMAN II [19, 20], HYPO [2, 3], CATO [1] and GREBE [8]. CBR systems in law are especially useful to give a situation-dependent interpretation of *open-textured concepts*. Legal rules typically contain abstract, open-textured antecedents. The rules have to allow application beyond immediate and known situations. The applicability of an open-textured

predicate to a new case can be analyzed by comparing the facts of the case to those of precedent cases in which the truth-value of the *ratio decidendi* was determined [4]. The technique of analogical reasoning allows applying an existing solution strategy to a wider class of problems, providing it is possible to construct a supporting correspondence mapping between partially enabled schemata and novel problems. This gives the opportunity to concretize an open-texture concept in a new context.

4 Case-Based Reasoning and Information Retrieval

Case-based reasoning (CBR) and information retrieval (IR) of decisions have much ground in common. The decisions are retrieved from a database of cases in order to help the legal professional to build an argument for his or her case at hand. In a CBR system the argument is automatically generated. But, CBR systems cope with the problem of the manual implementation of the case representations. As a result, CBR systems in law operate in restricted domains and are commercially not widespread. IR systems do not automatically construct the arguments, but retrieve documents or text passages in which the arguments must be found. They cope with the problem, that their search is often not very reliable returning documents that are not relevant or not returning all relevant documents, and especially returning too many potentially relevant documents that are impossible to consult.

In this framework an IR system that partly uses CBR technologies seems very useful in order to improve the search. Firstly, we need an intelligent automated indexing of the cases when storing cases in information retrieval systems that better allow for CBR strategies. Secondly, given the fact that court decisions are retrieved for helping the legal professional to construct arguments, the search operation might include a light form of CBR aiming at a better matching of the information need and the case representations and providing filtering of the cases.

4.1 Case Representations

Both CBR and IR systems use *indexing representations* that can be efficiently searched. These representations are made in a formalism that can be processed by a computer (such as vectors, rules, frames and semantic nets). In information retrieval simple inverted files (i.e., lists of key terms with the corresponding addresses of the documents in which they occur) are still very common.

CBR – including legal CBR - learns that the surface features of a case, i.e., the features that are most apparent such as the most obvious facts of the case, do not always reveal all the aspects of a case. Additional knowledge can be added to the case representations making them more suitable for reasoning and learning from it in new situations [11]. First of all, this refers to describing the cases at a more abstract level or describing the decisions with factors and legal issues. Also, extra knowledge with regard to the different contexts in which the cases are used, the reasoning steps followed in the cases, and other external knowledge sources that prove useful can be added [18, pg. 369 ff.].

In IR representing all the words of a text is popular because it is thought that these words still do have all the information in se –although this is a false assumption because we loose many of the relationships between words, phrases and sentences. Blair [6] discusses two competing forces when making document representations used in information retrieval. He compares the document representations based on their exhaustivity. *Exhaustivity* refers to the degree to which all the concepts and notions included in the text are recognized in its description. When the representations are highly descriptive (i.e., are very exhaustive) searches will

tend to have high recall and low precision. But, if descriptions are biased towards discrimination, i.e. having few indexing terms that very well discriminate one document from the other, searches will tend to have high precision and low recall. When only using discriminating descriptions, the searcher might not be able to anticipate any representations of relevant documents and recall and precision will both be zero.

Kolodner [18, pg. 280 ff.] equally discusses two competing forces. She compares the representations based on their specificity. *Specificity* refers to the degree of generalization of the representation. General indices are good at driving an initial retrieval and allow for discovering the deep basis for similarity that crosses detailed domains. On the other hand specialized representations are useful for refined and precise searches. Less specific indexing makes retrieval of partial matches easier but results in many largely inappropriate cases being retrieved; more specific indexing results in retrieval of fewer cases and more appropriate ones but makes retrieval of unanticipated partial matches more difficult.

When building case representations for an IR task, storing *all the words* (perhaps with the exception of stopwords) is good practice. In addition, *more general terms* or concepts (i.e., the factors and issues of a case) bridge the gap in searching the same concept over texts that contain different fact descriptions [13, 17]. This task is not without any problems. First of all, which subject categories, factors or issues should be assigned? According to Blair [6], their number is almost unlimited. There is both empirical as well as theoretical evidence that there may be no practical limit to the number of ways a single document can be described. Of course, in certain legal domains we can build useful domain classifications, thesauri and ontologies. A *thesaurus* portraits the semantic relationships that hold between the terms when they refer to different aspects of a common concept or domain [30]. The main relationships used in retrieval are the ones that define synonyms or that broaden terms (hypernyms) or narrow terms (hyponyms). A thesaurus provides a grouping or classification of the terms that have a certain semantic relationship into thesaurus classes. Thesaurus classes have a similar function as an *ontology* used in natural language processing. In legal information systems, great importance is attached to the availability of a thesaurus. However, not all legal thesauri are electronically available. This situation might improve considering the current interest in building an *ontology* for a specific legal domain (e.g., [31]). In addition, legal documents contain ordinary language. As a result concepts and their relationships from ordinary language should also be covered.

Another problem is the assignment of the concept descriptors. When done manually, it remains an impossible task, when done automatically there is the problem that existing thesauri or ontologies do not provide enough context to unambiguously assign the descriptors to the texts [5, pg. 77] and that the learning algorithms typically used in text classification do not always yield good results when used for decisions. Examples of training *text classifiers* upon court decisions are found in [9, 10, 28]. In addition, these techniques require examples that are manually classified. Learning classification patterns for text is currently heavily researched (e.g., the use of support vector machines) and future improvements of the techniques might yield better results.

Adding extra knowledge to the IR system with regard to all the *contexts* in which the cases might be searched is difficult for legal cases. At any time new interpretations or new contexts in which a case is useful can come up.

Adding a representation of the *reasoning* followed in a case, for instance by representing the rhetorical relationships between concepts and terms can be very useful for retrieval [21, 23]. The rhetorical structure of a text is a main indicator of how information in that text is ordered into a coherent informational structure. One could argue that in court decisions the rhetorical structure is even more important than in other texts, since it is in the nature of court

decisions to represent an argumentation. For instance, relationships of cause, evidence, contrast, elaboration etc. can occur between sentences and clauses. Rhetorical structure analysis could therefore be a very useful tool for locating important information and analyzing the structure of the argumentation in this type of texts. Unfortunately, little research is done with regard to a rhetorical structure analysis of decisions.

More sophisticated IR queries (see below: query by example and question answering) require attributes other than words and key terms. In addition, the representation can include the relationships between terms and their relationship to either plaintiff or defendant, positional information of the terms in the document (e.g., in which part, in which sentence, which position in the sentence), syntactical information (e.g., part of speech tag of each word) and semantic information (e.g., semantic class such as named entity and semantic role in the sentence such as the actor). The case representation should be at least as expressive as the natural language text of the decision [14]. We argue that the representation should be as rich as possible, but depending upon the type of search query different subsets of the representations should be used.

Case representations should be efficiently searched. They are usually stored on the servers of search engines. Rich case representations need a large storage space. The representations can be divided into *essential indices* which will be searched in main memory and *secondary index attributes* that can be stored in secondary memory and be put in main memory in case of more sophisticated queries. This is an approach that is currently researched in multimedia information retrieval (e.g., geographic/cartographic applications).

4.2 Similarity Computations and Analogical Inference

In computing the similarity between the case at hand and the precedent cases in CBR or between query and documents in IR, the context is very important. Certain cases are only relevant within a certain context.

Many current legal CBR systems often use a simple model of precedents consisting of representations of the material facts, factors and the outcome of precedents (e.g., pro plaintiff, pro defendant) together with a simple measure of similarity. The choice of facts, factors or issues and the choice of a rationale depend upon the context of the new case. There are specified rules about what is relevant and where and how to determine matches. Essentially, it is decided, a priori, what the relevant features of a case are, or are in certain contexts. When reasoning with legal cases, relevance and degree of sharing are difficult to be fixed with simple rules in advance.

Implementing all *context conditions* in advance is equally not possible in IR – especially not when retrieving court decisions, because of the almost infinite amount of circumstances in which court decisions can be searched. It is equally difficult to represent in a static way the search context, i.e., the context within which each person seeks information and which consist of cognitive, social and other factors related to a person's task, goals and intentions in advance. It is very difficult to provide a complete model of the manner in which knowledge of social customs, historical development of legal doctrine, or models of social justice can guide decisions about the equivalence of cases with respect to a legal category. In IR this is solved by an iterative search in which the query is automatically improved based upon relevance feedback of the user.

In CBR once similar cases have been found the new case receives the classification or solution of the matching cases. This is called analogical inference. Kolodner [18, pg. 475 ff.] makes a distinction between *exemplar-based classification* and interpretive case-based reasoning in making analogical inferences. In the former classification is done by finding the

exemplar most similar to the new item and assigning the new item its classification, which could result in finding similar cases with different classifications or outcomes. In *interpretive case-based reasoning* a new case is compared with the old ones and justifiable arguments are sought to support one or another decision. The inferences that are made can be subject to constraints. For instance, in Winston's view of analogy there must be causal links between the common information of the old and the new case and the information inferred by the analogy [32]. Two stories match when not only the facts match, but also the relationships between them.

A precedent case seldom is completely equal to the current case. The old situation cannot identically be transposed to the new situation. *Substitution* is the process of choosing and installing a replacement for some part of a situation. Again context is important: substitution is allowed within a certain context.

Exemplar-based classification is applied in the oldest case-based reasoning systems in law (e.g., MacKaay and Robillard, 1974 cited [16]). Technically, such an approach can be implemented with a *k-nearest neighbor classifier* (See [22, pg. 124 ff.] for an overview the use of this technique in text classification). The nearest neighbor is determined by a metric that typically consists of a weighted sum of feature differences (e.g., Euclidean distance) or similarities (e.g., cosine function) between the two cases that are compared. The legal classification of a new case is revealed by finding the new case's nearest neighbor in the feature space and applying the legal classification of that neighbor to the new case. This simple method works very well when the cases are represented with features that represent properly the rationale of a decision. However, often there are multiple possible rationales of a case (i.e, other contexts of using the case) (cf. [8, pg. 56]). Exemplar-based classification can be computed both with raw facts and more abstract concepts. When case features are represented with a hierarchy of facts, factors and issues, cases are compared with a similarity metric that takes into account this hierarchical representation. The degree of similarity is then, for instance, computed in terms of the most specific common abstraction. The more specific the most common abstraction, the better the match is.

Interpretive case-based reasoning is very much valued in the legal domain. The HYPO system is a nice example hereof [2, 3]. The main purpose of HYPO is to produce arguments by analogical inference and to identify issues that may be used by a lawyer, rather than to find which particular case applies to the current case and the outcome of which should be followed. HYPO compares cases based on shared factors. The most-on point cases, i.e., the precedent cases that share many factors with the new case that have the desired outcome are used to create arguments for the current case. Cases that are more or less on point, but have an undesired outcome are used to make counter-arguments.

Related to the view of Winston [32] is the finding of the similarities between cases based on a sharing of facts and of the reasoning. This model requires an explicit representation of the reasoning under which a decision follows the material facts of a case. This approach is used in the GREBE system [8].

In IR systems, similarity between query and cases is found by direct comparison of the raw facts or by comparing concepts. The *substitution* approach is very common. In a retrieval context, the query representation does not always match a document representation although the document is relevant for the query. A form of transformation of either the query or the document representation is needed to allow the matching. When matching text, simple transformations that improve the match are removal of inflectional morphemes (lemmatization) and stemming (removal of inflectional and derivational morphemes). The terms of both queries and documents can be expanded or replaced by synonyms or related terms obtained from a thesaurus or ontology or learned from a training corpus.

Another example of transformations in IR is the *logic-based retrieval model* [29]. This model will use the information in query and document in combination with domain knowledge, linguistic knowledge, and knowledge of users' interests and strategies from a coded knowledge base. Ranking according to relevance then depends upon the number of transformations necessary to obtain the matching and the credibility of the transformations. To represent uncertain implications and reason with them, modal logic is sometimes used [24]. For instance, when a matching between query and text representation is not successful, the text representation is transformed in order to satisfy other possible interpretations (cf. the possible worlds of modal logic) that might match the query.

Two newer forms of retrieval models also fit the CBR paradigm. They are query by example and question answering and are currently heavily researched.

In *query by example* retrieval the searcher provides an example object and the system retrieves similar objects, possibly ranked by decreasing similarity. This technique is commonly used in multimedia information retrieval, for instance, for the retrieval of similar images given an example image or similar musical melodies given example melodies. In case of text based information retrieval the technique is less widespread, but the retrieval of court decisions seems a nice application. There are different possibilities of finding a good example (e.g., description of the fact patterns, exemplary passage selected in a previous step in the search, the complaint that identifies the issues and circumstances of being litigated, the depositions given during the pre-trial of the lawsuit, and a case retrieved from a CBR system such as in the SPIRE system in [12, 25]). These exemplary documents are useful when they describe or exhibit the intellectual structure of a particular field of interest. In doing so, they provide both an indexing vocabulary and, more, importantly, a narrative context in which the indexing terms have a clearer meaning [7]. Such pieces of text often explicitly mention the relationships between some or all of the issues or topics they identify (e.g., that certain topics are related causally; or how specific events are related chronologically).

A query by example model is also one step closer in finding cases that suggest a new interpretation of an open-texture concept, although we acknowledge that this represents a difficult task to be performed automatically.

In a *question answering* system a searcher poses a real question in natural language and the system does not retrieve the documents in which the answers can be found, but the answer to the question (e.g., Which sum is allocated to social securities in the budget for 2003?) (cf. [27]). Single questions are automatically answered by using a collection of documents as the source of data for the production of the answer. The advantage of using a question answering search is similar to a query by example search: the question provides context for the meaning of the search terms and the question includes linguistic relationships between the terms that should also be found in the documents. If the searcher were able to replicate rhetorical reasoning, he would have a powerful searching capacity. As he searched, he would be following the cognitive process of devising an argument. Even, if the technology only is used to identify passages or sentences in which the answer can be found, it is already a useful tool for the retrieval of court decisions.

5 Summary of the Findings

Case-based reasoning can learn us how to improve current systems that store and retrieve court decisions. IR systems should automatically acquire case representations that contain more information than just the words of the text. Because, we can never predict in advance what information in the cases will be relevant to search, the representations should be as close as possible to the natural language text, so that many inferences from the text are possible.

Promising directions include the application of text classification techniques and shallow linguistic analysis of the texts. We think that rhetorical structure analysis of the decisions is very useful to research. A solid organization of the representations must guarantee that they can be searched efficiently.

Having exhaustive case representations, IR needs finer techniques for matching. Matching algorithms must be able to distinguish which features of the representations to focus on when different types of queries (e.g., query by example and question answering) are used. These newer forms of retrieval will probably help us sifting through many cases when querying large databases. We have outlined a number of research tracks that can be implemented and tested. We hope that these research directions will be followed in future projects.

References

[1] Aleven, V., Teaching Case-Based Argumentation Through a Model and Examples.Ph.D. Dissertation, University of Pittsburgh, 1997

[2] Ashley, K., Modeling Legal Argument: Reasoning with Cases and Hypotheticals.MIT Press, Cambridge, MA, 1990.

[3] Ashley, K. and Rissland, E.L., A Case-Based Approach to Modeling Legal Expertise. *IEEE Expert,* **3** (3) (1988), 70-77.

[4] Bing, J., The Problem of Finding a Precedent. In: Nagel (ed.), *Law, Decision-making and Microcomputers*. Quorum Books, New York, 1991, pp. 285-307.

[5] Bing, J. and Harvold, T., Legal Decisions and Information Systems.Tano, Norway, 1977.

[6] Blair, D.C., The Challenge of Commercial Document Retrieval. Part I: Major Issues, and a Framework Based on Search Exhaustivity, Determinacy of Representation and Document Collection Size, *Information Processing and Management*, **38** (2002) 273-291.

[7] Blair, D.C. and Kimbrough, S.O., Exemplary Documents: A Foundation for Information Retrieval Design, *Information Processing and Management*, **38** (2002) 363-379.

[8] Branting, L.K., Reasoning with Rules and Precedents: A Computational Model of Legal Analysis. Kluwer Academic Publishers, Boston, MA, 2000.

[9] Brüninghaus, S. and Ashley, K.D., Finding Factors: Learning to Classify Case Opinions under Abstract Fact Categories. In: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*. ACM, New York, 1997, pp. 123-131.

[10] Brüninghaus, S. and Ashley, K.D., Toward Adding Knowledge to Learning Algorithms for Indexing Legal Cases. In: *Proceedings of the Seventh International Conference on Artificial Intelligence and Law*. ACM, New York, 1999, pp. 9-17.

[11] Carbonell, J., Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition. In: S. Michalski, J. Carbonell and T. Mitchell (eds.), *Machine Learning* 2. Morgan Kaufmann, San Francisco, CA, 1986, pp. 371-392.

[12] Daniels, J.J. and Rissland, E.L. Finding Legally Relevant Passages in Case Opinions. In: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law,* ACM, New York, 1997, pp. 39-46.

[13] Dick, J.P., Conceptual Retrieval and Case Law. In: *Proceedings of the First International Conference on Artificial Intelligence and Law*. ACM, New York, 1997, pp. 106-115.

[14] Dick, J.P., Representation of Legal Text for Conceptual Retrieval. *Proceedings of the Second International Conference on Artificial Intelligence and Law*. ACM, New York, 1999, pp. 244-253.

[15] Gardner, A., An Artificial Intelligence Approach to Legal Reasoning. MIT Press, Cambridge, MA, 1987.

[16] Gray, P. N. , Artificial Legal Intelligence. Dartmouth, Aldershot, UK, 1997.

[17] Hafner, C., Conceptual Organization of Case Law Knowledge Bases. In: *Proceedings of the First International Conference on Artificial Intelligence and Law*. ACM, New York, 1987, pp. 35-42.

[18] Kolodner, J. , Case-Based Reasoning.Morgan Kaufmann, San Mateo, CA, 1993.

[19] McCarty, L.T., An Implementation of Eisner v. Macomber. In: *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*. ACM, New York, 1995, pp. 276-286.

[20] McCarty, L.T. and Sridharan, N.S., The Representation of an Evolving System of Legal Concepts II: Prototypes and Deformations. In: *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, San Mateo, CA, 1981, pp. 246-253.

[21] Mital, V. and Johnson, L., Advanced Information Systems for Lawyers.Chapman and Hall, London, 1992.

[22] Moens, M.-F., Automatic Indexing and Abstracting of Document Texts (The Kluwer International Series on Information Retrieval 6). Kluwer Academic Publishers, Boston, MA, 2000.

[23] Moens, M.-F. and De Busser, R., First Steps in Building a Model for the Retrieval of Court Decisions, *International Journal of Human-Computer Studies*, in press.

[24] Nie, J.-Y., Towards a Probabilistic Modal Logic for Semantic Based Information Retrieval. In: N.J. Belkin, P. Ingwersen and A.M. Pejtersen (eds.), *Proceedings of the Fifteenth SIGIR Conference*. ACM, New York, 1992, pp. 140-151.

[25] Rissland, E.L. and Daniels, J.J., Using CBR to Drive IR. In: *Proceedings of the International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, San Francisco, CA, 1995, pp. 400-407.

[26] Rissland, E.L., Skalak, D.B. and Friedman, T.M., BankXX: Supporting Legal Arguments Through Heuristic Retrieval, *Artificial Intelligence and Law*, **4** (1) (1996) 1-71.

[27] Svoboda, W.R., Structuring Analysis of the Law for the Purpose of Installing Query-Answering Systems. In: C. Ciampi (ed.), *Artificial Intelligence and Legal Information Systems*. North Holland, Amsterdam, 1982, pp. 181-191.

[28] Thompson, P., Automatic Categorization of Case Law. In: *Proceedings of the 8^{th} International Conference on Artificial Intelligence and Law.* ACM, New York, 2001, pp. 70-77.

[29] Van Rijsbergen, C.J. , A Non-Classical Logic for Information Retrieval, *The Computer Journal*, **29** (1986) 111-134.

[30] Wang, Y.-C. and Vandendorpe, J., Relational Thesauri in Information Retrieval, *Journal of the American Society for Information Science*, **36** (1) (1985) 15-27.

[31] Winkels, R., Bosscher, D., Boer, A. and Hoekstra, R., Extended Conceptual Retrieval. In: J. Breuker, R. Leenes and R. Winkels (eds.), *Legal Knowledge and Information Systems*. IOS Press, Amsterdam, 2000, pp. 85-97.

[32] Winston, P., Learning and Reasoning by Analogy, *Communications of the ACM*, **23** (1980) 689-703.