

Thesaurus-based Retrieval of Case Law

Michel KLEIN^a, Wouter VAN STEENBERGEN^b, Elisabeth M. UIJTENBROEK^b,
Arno R. LODDER^b and Frank VAN HARMELEN^a

^a *Department of Artificial Intelligence, Vrije Universiteit, The Netherlands*

^b *Faculty of Law, Vrije Universiteit, The Netherlands*

Abstract. In the context of intelligent disclosure of case law, we report on our findings on methods for retrieving relevant case law within the domain of tort law from a repository of 68.000 court verdicts. We apply a thesaurus-based technique to find specific legal situations. It appears that statistical measures of term relevance are not sufficient, but that explicit knowledge about specific formulations used in law and case law are required to distinguish relevant case law from irrelevant. In addition, we found out that the retrieving legal concepts with an “interpretive” character requires a different method than concepts do not require additional interpretation.

Keywords. retrieval, case law, ontology, thesaurus

1. Introduction

The judiciary is faced with enormous case loads. Alternative dispute resolution mechanisms such as mediation can help to reduce this workload. Mediation is not always popular (if known at all), in particular since litigants are often not aware of their chances in court, and normally overestimate their chances. In the BEST-project¹ [5] we strive to provide disputing parties with information about their legal position in a liability case. We are developing a system that supports users by retrieving relevant case law, i.e. court decisions. In this way parties are given the opportunity to form a judgment about whether they could hold another party liable for certain caused damage or if they could be held liable themselves. Also, parties can determine how much room for negotiation is available when settling the damage. By information about previous court decisions, where relevant taking into consideration other factors such as time, costs, emotions, etc., a well-rounded impression is obtained about a parties’ BATNA (Best Alternative To a Negotiated Agreement), that is: the best option a party has if negotiation fails [1].

An important problem we have to face in this context is the discrepancy between the terminologies used. At least three different vocabularies can be distinguished: the vocabulary that laymen use to describe their case, the terminology found in legislation, and the wording in court decisions. To address this problem, we use a combination of statistical text retrieval methods and knowledge-based techniques. The basic idea is to de-couple the task of creating a meaningful and complete description in legal concepts of the case at hand (1) and the task of retrieving similar cases (2).

¹<http://www.best-project.nl>

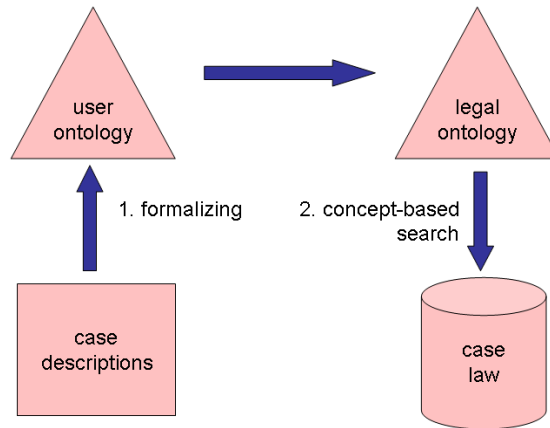


Figure 1. General architecture of the BEST system

Figure 1 shows a conceptual architecture that reflects the principle of de-coupling case description and document retrieval that is the basis for the work in the BEST-project. The case description explicates relevant aspects of the case at hand using a structure and terms provided by a user ontology. This ontology is mapped on a second conceptual structure that is used to index case law.

In this paper, we report about the second task: retrieving case law in which a specific legal case is described. For this we use thesaurus-based statistical retrieval technique[2]. This technique uses a thesaurus to create a vector representation of each document. Documents can be compared by comparing their vector representations. Searching is performed by creating a “query document” and comparing the vector representation of this query with the vector representations of the other documents. The two questions we address in this paper are: *is this technique suitable for retrieving case law in which a prototypical legal case is described* and *how to create—for a prototypical legal case—a search document that can be used to retrieve documents in which similar legal cases are described*.

In the remainder of this paper we first describe the thesaurus-based retrieval technique that we use. Section 3 contains the research questions in detail, the experimental setup and the results. In Section 4 we discuss the observations that can be made from the experiments. We finish the paper with some conclusions about the characteristics of retrieving case law and future work.

2. Concept-based retrieval

For the retrieval of the relevant documents, we use a thesaurus-based statistical indexing method. This technique has been implemented in a commercially available software tool by Collexis BV.² The main advantage of this technique is that, when compared to standard information retrieval techniques based on the *vector space model* [3], the indexing is guided by a thesaurus; in this way, only terms relevant in a specific domain are taken into account.

²<http://www.collexis.nl>

The indexing method works as follows [6,7]. The indexing algorithm first detects sentences in documents and removes stop-words from the sentences. After this it normalizes the remaining words, which means that nouns are reduced to the singular form and verbs to the first person singular form. In our experiments, we have used a specialized normalization engine for the Dutch language for this. From these normalized terms or phrases, the relevant ones are then identified using a domain-specific thesaurus.

A list of the relevant concepts identified in a document is called a *concept fingerprint* of that document. For each identified concept a unique concept identifier is added to the fingerprint. This concept identifier is assigned a relevance score, based on term frequency and the specificity of the term in the thesaurus (which is the depth in the hierarchy), and the lexical similarity of the term with the textual contents [4].

A fingerprint can be seen as a vector in a high dimensional space. The dimensions of this space are formed by the concepts of the thesaurus. The weight (or value) in each dimension is the relevance score for the concept in the document.

The search is performed by a matching engine in the software, which matches a search vector with the vectors of the indexed documents. The vector for the search query is calculated in a similar way as described above. The matching engine will compute the distance between the query vector and the vectors of the documents. The result of the matching engine will be a set of document vectors sorted on their distance to the query vector. This is presented to the user as a ranking on relevance of the indexed documents.

3. Retrieval experiments

3.1. Data sources

The case law database used to disclose similar cases, is that of the public website www.rechtspraak.nl. For processing purposes we have all available 68.000 cases locally stored. Given the over 1 million legal verdicts annually, this is a low number. Nonetheless, this database contains almost all digitally available newer case law in the Netherlands. The verdicts have some meta-data attached to them, e.g. the type and location of the court, the date of the verdict, a nationally unique identifier and for around 50% of the verdicts (the newest) a summary of a few sentences. Internally, the documents have no computer-parsable structure, but are plain text instead.

3.2. Research questions

As stated in the introduction, we want to answer the following questions with our experiments:

1. is concept-based search as described above a suitable technique for retrieving case law in which a prototypical legal case is described?
2. how to create—for a prototypical legal case—a search document that can be used to retrieve documents in which similar legal cases are described?

There are several possible answers to these two questions. With respect to the second question, the creation of a search document, we investigate three different options:

- A. distill relevant terms for the specific legal situation from the articles from the code (we call this: *code-based fingerprints*);

- B. manually describe relevant terms for the specific legal situation, based on a analysis of the terminology used in the case law (*case-based manually created fingerprints*);
- C. select a number of relevant cases and use these documents together as one search query (*case-based automatically generated fingerprints*).

Our hypothesis is that they will perform in increasing order. First, this is because we expect that articles from the code use different terminology than the terminology used in case law, therefore case-based fingerprints will be better suited to identify relevant cases. Second, we expect that the indexing process for the *generated fingerprints* will automatically distinguish the most important terms and therefore perform better than the manually created fingerprints.

With respect to the third question, there is not such a clear list of options. We have to decide about the *size* and *scope* of the thesaurus and the *type of vocabulary* used, i.e. terms from law text, general legal terms, or terms specific to case law.

3.3. *Experimental setup*

We started with making a selection of legal cases for which we want to identify relevant case law. We chose three fairly different types of liability and a fourth one that is very close to one of the other situations:

- liability for misleading advertisements;
- liability for non-subordinates;
- liability for real estate;
- liability for subordinates.

The reason for this choice is twofold. First, by starting with very diverse legal cases, we can check whether our set up is suited for distinguishing legal cases at all. Second, the broad choice of situations will prevent us drawing conclusions from a not representative subset of the liability field.

The idea behind the two similar situations (liability for subordinates / non-subordinates) is that this will learn us whether the technique is also able to distinguish different legal cases that are quite similar to each other.

For each of these legal cases the Dutch Civil Code contains a specific section. For each of the selected legal cases we did the following three things:

1. from the database with court decisions we selected a number of cases that are relevant;
2. we distilled all relevant terms from the law text;
3. we analyzed relevant cases and made a list of important terms used.

In addition, we have created a thesaurus with legal terms. The thesaurus is manually created from the terms identified in task 2 and 3 in the list above and the terms identified in the law text for other types of liability. This resulted in a thesaurus with 360 concepts. The structure of this thesaurus is imposed by the structure of the law itself, i.e. “liability” is the root concept with more specific types of liability below it, e.g. “liability for persons”, which in turn has “liability for subordinates” below it. The relevant terms are placed below the types of liability for which they hold, including some synonyms.

The thesaurus is used to index the data set (i.e. creating fingerprints for each document in the repository) and the other material is used to create different search documents for which fingerprints are calculated. We then used the search documents' fingerprints to search for relevant cases. The top of the highest ranked results were evaluated on relevance.

3.4. Experiments and results

3.4.1. A: Code-based fingerprints

In a first set of experiments, we evaluated the code-based fingerprints, i.e. the fingerprints with terms distilled from the articles from the code. We expected not very good results here, as we assumed that the vocabulary used in the cases is different from the vocabulary in the articles from the code. As can be seen in Table 1, the correctness figures for the 10 highest ranked are indeed quite low. For two fingerprints, this set did not contain any relevant result at all. In the other one, we only found 3 really relevant cases, but also two cases in which the article searched for was only casually mentioned. Note that for article 6:170, we found 8 slightly relevant cases.

Article	Correctness	
6:170 subordinates	1 / 10	10 %
<i>(including slightly relevant)</i>	9 / 10	90 %
6:171 non-subordinates	0 / 10	0 %
6:174 real estate	3 / 10	30 %
<i>(including slightly relevant)</i>	5 / 10	50 %
6:194 misleading advertisement	0 / 10	0 %
6:162 unlawful act	10 / 25	40 %
<i>(including slightly relevant)</i>	12 / 25	48 %

Table 1. Correctness figures for code-based fingerprints.

3.4.2. B: Case-based manually created fingerprints

We did the same experiment for case-based manually created fingerprints—fingerprints based on important terms identified by the expert in a selection of the case law. This resulted in the figures printed in Table 2. For two of the three articles the results are fairly good. For one article, the results are not so good; interestingly, this is an article for which a good result was obtained for the code-based fingerprints.

Article	Correctness	
6:171 non-subordinates	5 / 10	50 %
<i>(including slightly relevant)</i>	7 / 10	70 %
6:174 real estate	1 / 10	10 %
6:194 misleading advertisement	10 / 12	83 %
<i>(including slightly relevant)</i>	11 / 12	92 %

Table 2. Correctness figures for the case-based manually created fingerprints.

3.4.3. C: Case-based automatically generated fingerprints

Thirdly, we evaluated the performance of automatically generated fingerprints—fingerprints based on the full text of a set of pre-selected relevant cases. We started with fingerprints based on 5 case descriptions for 3 different legal cases.

The results vary for the different articles from the code (see Table 3). The table lists the number of cases used to create the fingerprint, the number of relevant cases as fraction of the total number of evaluated cases, and this fraction represented as a percentage.³ We evaluated the relevance of the first 15 documents returned, but we did not count the documents that were used to *create* the fingerprint. This explains the difference in the totals in the column with the correctness.

Article	# Cases	Correctness	
6:170 subordinates	5	7 / 10	70 %
6:171 non-subordinates	5	0 / 10	0 %
6:174 real estate	5	3 / 8	37 %
6:194 misleading advertisement	5	10 / 13	77 %
6:171 non-subordinates	20	2 / 10	20 %
6:162 unlawful act	249	8 / 25	32 %
(including 6:174)		13 / 25	52 %

Table 3. Correctness figures for case-based automatically generated fingerprints.

A hypothetical explanation for the diverse results is that the sets of documents from which the fingerprint are generated are too small. To check this, we generated a fingerprint from a larger set of documents (20 cases) for the worst performing legal case, i.e. “real estate”. Because the total number of cases in the data set for these situations are between 40 and 100, it is in practice not realistic to base fingerprints on much more than 20 fingerprints. As can be seen in the table, the results were still not good: only 2 out of the 10 cases highest ranked were relevant.

Finally, we have generated a fingerprint from a very large set of documents. We used the general “unlawful act” article 6:162 for this, as this is the only article for which we had enough case law (around 500) to do this experiment. The fingerprint for article 6:162 is based on 249 cases. To our surprise, the results were still disappointing: only 8 from the 30 highest ranked cases were relevant and not yet used to create the fingerprint. Even when the cases about the related article 6:174 were considered as relevant, we only count 13 cases. Moreover, the first case which was not part of the fingerprint appeared to be irrelevant.

3.4.4. D: Adding legal phrases

While looking for an explanation for the results of the previous experiments, especially the under-performance of the fingerprint for “liability for real estate”, we considered

³The percentages in this and the following tables are only provided to give a comparable indication of the quality of the results, but should not be interpreted as precision measures. This is because of two reasons: first, the number of evaluated cases is too low, and second, the total number of evaluated cases differs for the different experiments. The result is always a ranking of all documents and there is no straightforward way to determine a threshold were to stop evaluating. In practice, we stopped when we had several irrelevant cases in succession.

that the wide variety facts of the case probably blurred the legal similarity. For example, “liability for real estate” copes with all kinds of real estate, including roads, and accidents with all kinds of vehicles because of shortcomings in the road. However, we also found out that there are *typical phrases* that are used to prove a specific type of liability.

Therefore, we extended the case-based manually created fingerprints with such phrases. We distinguished the different argumentation lines that can be followed to prove something and the typical phrases that were used in the argumentation. On average, we added around eight phrases per legal case. Translated examples of such phrases for “real estate” are: “causing danger for persons or objects”, “owner of a property” and “requirements that in a given situation”. We have added these phrases also to the thesaurus and re-indexed the complete repository.

The results of this experiment are listed in Table 4. The figures indicate that there are more relevant cases returned than in previous experiments. What is also interesting, but not visible in the figures, is that the ordering seems to be better than in previous experiments: the relevant and irrelevant cases were less intermixed than before. In this experiment we also counted the number of relevant cases that did not explicitly mention the article number. These are interesting cases because these are found by relevant wording only, and not because the article number is mentioned.⁴ As can be seen, there are at least a few cases that are relevant, but do not literally contain the article number.

Article	Correctness		Not explicit
6:170 subordinates	22 / 25	88 %	2
6:171 non-subordinates	7 / 15	47 %	0
<i>(including slightly relevant)</i>	8 / 15	53 %	0
6:174 real estate	24 / 30	80 %	1
<i>(including slightly relevant)</i>	26 / 30	87 %	1
6:194 misleading advertisement	17 / 21	80 %	3

Table 4. Correctness figures for the case-based manually created fingerprints including legal phrases.

3.4.5. E: Removing the thesaurus

Finally, we redid the last experiments with no other concepts in the thesaurus, i.e. we reduced the thesaurus to the four different types of liability and their relevant phrases. This resulted in a thesaurus of 25 concepts expressed in 50 terms (i.e. 25 synonyms).

When using this thesaurus to index the complete data set, around 7000 documents (out of 68.000) were ignored because none of the terms in the documents were similar to terms in the thesaurus. The remaining documents were indexed with only 1.08 terms on average. This suggests that sensible results are unlikely, because it almost means that for each document a single keyword is attached.

The correctness figures for this method were still quite high, 70% for the first 10 hits in for “sub-ordinates liability”. However, all of them literally contained the article number. As we have seen in one of the previous experiments, there are also relevant cases in which the article number is not literally mentioned.

⁴The mere fact that the article number is mentioned is not enough to be a relevant case, of course.

4. Discussion

Several observations can be made from our experiments. First, we noted that only for one legal case (“liability for misleading advertisements”, 6:194) the results for the automatic case-based generated fingerprints were notably better than the code-based fingerprints. A possible explanation is that the specific code text uses very abstract formulations, which have only a few terms in common with actual cases. We noted that the code specifies a non-exhaustive list of the type of statements that can be seen as misleading (“statements about the contents”, “statements about the amount”, etc.). The terms used in this list of typical misleading statements will not frequently occur, as they describe statements at an abstract level. These abstract terms are different from the concrete terms that are used in the case law. Thus, even although the case law will contain the term ‘misleading advertisement’ very often, the resulting fingerprint will be quite different. The automatically generated fingerprints from the cases do contain the concrete terms from the cases, of course.

A second interesting observation is that when using code-based search, we found for some of the legal cases (e.g. 6:162, 6:170 and 6:194) many *indirectly relevant* cases, i.e. cases in which the article was only casually mentioned. This finding can possibly be explained by the *interpretive* characteristic of the legal concepts mentioned in the code for these articles. When such concepts are not precisely defined the legislator intentionally left room for interpretation by judges. Legal reasoning that involves interpretation, is a manifestation of the application of a vague concept. A vague concept is characterized by its open texture. An example of such a vague concept is ‘the reasonable man’ or ‘sufficient’. In situations where vague concepts are used additional case law has to be formed to understand the precise meaning of this code. Cases about other law articles in which the same interpretive concept is used often refer to this case law, causing a lot of indirectly relevant results. According to this argumentation, a high number of *indirectly relevant* cases would be a signal of code text with many interpretive concepts.

Another observation, which is not directly visible in the figures, is that the analysis of the results showed that the type of cases returned for the automatic case-based fingerprints and the code-based fingerprints are very different for articles 6:171, 6:174, 6:162, although the percentages of correctness are comparable. Code-based fingerprints resulted in cases that literally contained some non-interpretable concepts in code law, while case-based fingerprints resulted in cases that define the meaning of interpretive concepts in code. This suggests that code-based fingerprints are useful for finding *non-interpretable concepts*, e.g. concepts that have a precise meaning in the law, while case-based fingerprints are more useful to find *interpretive concepts*. This is in agreement with the intuition that the meaning of interpretive concepts is defined by case law.

Another interesting finding is that manually created fingerprints in general perform better than automatically generated fingerprints (except for one of our examples, i.e. “real estate”). This is contrary to what we expected. This might have to do with the large number of real world situations in which some legal concepts can be relevant. If there are many situations, there are much more different terms used in the case descriptions. It is therefore more difficult to distinguish them by looking at the terms used. This is in particular a problem for the automatic method, as it uses the the number of occurrences of the terms as the measure to calculate the relevance. When manually creating fingerprints the most irrelevant terms are probably left out.

Finally, we have seen that adding typical legal phrases help a lot to improve the results. There are more relevant cases returned and the distinction between relevant and irrelevant seems to be crisper. However, the phrases alone are not sufficient. It seems that the phrases help to eliminate irrelevant cases in the top of the ranking (improve precision), but that additional concepts in the thesaurus are required for finding relevant documents that do not contain the literal article number (improve recall). An hypothesis is that the phrases are especially helpful for retrieving the concepts that need additional interpretation, i.e. the vague concepts.

5. Conclusions and outlook

In the introduction we have listed two research questions. We are now able to (partially) answer them. With respect to the first question, we can conclude that the thesaurus-based statistical retrieval technique can be used for the retrieval of relevant case law, but that the way in which the search document should be build depends on the character of the text in the code.

About the building of search documents (second question) we found out that it is not possible to conclude that either case-based documents or code-based documents performs better. Instead, it depends on the extent to which the text of the code contains “interpretive” concepts. Retrieving legal cases in which such vague concepts are used works better with case-based search documents, while retrieving legal cases which do not need additional interpretation performs best with search documents based on case law.

We also discovered that explicit knowledge about specific formulations used in law and case law helped a lot to improve the results. This brings us to the conclusion that statistical methods that use vector-based distance measures on itself are not sufficient for retrieving case law. Explicit knowledge about specific formulations used in law and case law, e.g. conditions to accept liability, are required. Our intuition is that this is because of the “hidden” characteristic of the concepts for which we search: specific legal cases are hidden under very different real world situations. We think that this feature is quite specific for the legal domain, and less prominent in other domains, e.g. the medical domain.

In addition, the analysis of the results helped us to formulate some hypotheses that we can investigate in future research. In the future, we will further investigate the best way to create a search document; we will carefully design search documents for each different legal case, using both code-based terms and phrases, based on an analysis of the specific situation. We will also work on methods to determine the threshold between relevant and irrelevant results, which can be used to calculate precision and recall measures.

Acknowledgements

This research was supported by the Netherlands Organisation for Scientific Research (NWO) under project number 634.000.436. We would like to thank Collexis B.V. for their permission to use their software, and the Dutch Council for the Judiciary (*Raad voor de Rechtspraak*) for allowing us to use their database with case law.

References

- [1] Berend R. de Vries, Ronald Leenes, and John Zeleznikow. Fundamentals of providing negotiation support online: the need for developing batnas. In John Zeleznikow and Arno R. Lodder, editors, *Second international ODR Workshop (odrworkshop.info)*. Wolf Legal Publishers, 2005.
- [2] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.
- [3] Gerard Salton and Chris Buckley. Term weighting approaches in automatic text retrieval. Technical report, Ithaca, NY, USA, 1987.
- [4] C. Christiaan van der Eijk, Erik M. van Mulligen, Jan A. Kors, Barend Mons, and Jan van den Berg. Constructing an associative concept space for literature-based discovery. *J. Am. Soc. Inf. Sci. Technol.*, 55(5):436–444, 2004.
- [5] Ronny van Laarschot, Wouter van Steenberghe, Heiner Stuckenschmidt, Arno R. Lodder, and Frank van Harmelen. The legal concepts and the layman’s terms. In *Proceedings of the 18th Annual Conference on Legal Knowledge and Information Systems*, Brussels, December 8-10 2005. IOS Press.
- [6] E.M. van Mulligen, M. Diwersy, M. Schmidt, H. Buurman, and B. Mons. Facilitating networks of information. In *Proceedings of the AMIA Symposium*, pages 868–872, 2000.
- [7] E.M. van Mulligen, C. van der Eijk, J.A. Kors, B.J. Schijvenaars, and B. Mons. Research for research: tools for knowledge discovery and visualization. In *Proceedings of the AMIA Symposium*, pages 835–839, 2002.