Legal Knowledge and Information Systems
M. Palmirani (Ed.)

doi:10.3233/978-1-61499-935-5-121

Analysing the Impact of Legal Change Through Case Classification

Roos SLINGERLAND, Alexander BOER & Radboud WINKELS[1]
Leibniz Center for Law, University of Amsterdam, Netherlands

Abstract. In this paper an automated solution for finding cases for analysing the impact of legal change is proposed and the results are analysed with the help of a legal expert. It focuses on the automatic classification of 15,000 judgments within civil law. We investigated to what extent several machine learning algorithms were able to classify cases 'correctly'. This was done with accuracies around 0.85. However, the data were scarce and the initial labelling not perfect, so further research should focus on these aspects to improve the analysis of the impact of legal change.

Keyword. Automatic classification, case-law, law changes.

1. Introduction

A regulation that involves all Member States of the EU is the Brussels I Regulation, which is a set of rules about the jurisdiction, recognition and enforcement of judgments in civil and commercial matters involving individuals resident in different Member States of the European Union and the European Free Trade Association (EFTA). The EU Regulation (EC) 44/2001, as the regulation is officially called, was created by the Council of the European Union and came into force in March 2002. However, the Court of Justice of the European Union (CJEU) stated that Article 23 regarding application of jurisdiction was too concise and therefore suggested a recast. Hence, a recast was created by both the Council and the European Parliament. It was implemented in 2015 and is officially called Regulation (EU) No 1215/2012. The main difference between the old and new regulation is that in the recast the rules of Brussels I were extended to defendants not domiciled in a Member State of the EU.

Although this recast was supposed to solve the shortcomings of the 44/2001 regulation, Danov [1] states that the application of this recast has been largely overlooked by both policymakers and literature. Since the regulation is still new and opinions differ as to its usefulness, more insight in its usage would be of great value for legal professionals and the EU. This research is complicated because there are no agreements regarding the referencing of the regulation or the recast in case decisions. Therefore these cases are hard to find manually. The use of automated tools could help.

This paper describes research on text analysis of cases involving the Brussels I Regulation and will discuss to what extent it is possible to design a supervised

[1] Corresponding author, PO Box 1030, 1000 BA Amsterdam, Netherlands, winkels@uva.nl

classification system that uses judgments of civil law cases from the Dutch portal to distinguish:

1. cases about Brussels I Regulation from all the other civil law cases;
2. cases from the Brussels I Regulation Recast from the Brussels I Regulation.

Answers to these questions will indicate whether or not a system could be used to reliably distinguish cases involving the Regulation or Recast after which further (manual) research is possible. The method of binary classification could then be extended to other Member States or even to other regulations that were changed. The system could then be used by experts at the start of a new law to assess its effects and impact. It is expected that because the second classification question has to deal with a smaller set of data, those results will score lower on accuracy than the results of the first classification problem.

The rest of this paper is organized as follows: We will first give a short overview of text classification in the legal field and describe an earlier attempt at automatic classification of civil law cases. Next we will discuss our research method and describe the two classifiers we built and evaluated. We will end with a discussion, conclusions and future work.

2. Text Classification

Text classification is a problem that has been studied in many domains, including the legal domain. Bruninghaus and Ashley [1] explain this demand by the desire of attorneys to find the most relevant cases and argue that this information need caused the wide interest of classification in the legal domain. De Maat e.a. [3] for example describe a study about the classification of legal sentences and the comparison of machine learning techniques against knowledge based classification. Goncalves and Quaresma [4] applied multiple algorithms to European legal texts and stated that legal texts are very suitable for text classification because of the unstructured format of the data. Bag-of-words method was used, but also part of speech tagging and lemmatisation were applied. They note the shortcomings of the bag-of-words method - the method being too simplistic to obtain good results - which was also mentioned by [1] and [3]. Above that, the researchers state that the legal language has a unique style and that the vocabulary and word-distributions differ from ‘regular’ English. All authors also point at the need for a pre-tagged training set and the difficulty of obtaining one. It is tedious and hard work and legal experts are busy and expensive.

Besides picking the right algorithms, proper feature selection is of great importance. Not only is it necessary to make large problems computationally efficient, but it can improve the accuracy substantially [5]. This increase of accuracy could also mean that less data is needed to obtain good results, which is a big advantage for a system.

2.1 An earlier attempt

Zheng [6] made an analysis of a data set obtained from the Dutch portal rechtspraak.nl for cases until October 2016 and used the MAchine Learning for LanguagE Toolkit (Mallet). The Dutch portal for case law contains a small, but growing part of all judicial decisions in the Netherlands. Case citations in these decisions are sometimes explicitly marked in metadata (e.g. the first instance case); references to legislation only the main

one(s) in recent cases. The texts are available in an XML format, basically divided in paragraphs, with a few metadata elements. The court decisions do not contain inline, explicit, machine readable links to cited legislation or other cases. So even when the metadata contain such references, we do not know in which paragraph the case or article was cited, nor how often.

First, an indication of the field of law for our purpose was made: 'civil law' is the most common in both the old (77%) and new (73%) regulation. Topic modelling was used to generate multiple topics and then multiple classifiers were trained and tested. For the old regulation an accuracy of 0.64 was obtained, and for the recast an accuracy of 0.78.

There are several differences between this earlier work and the research reported in this paper. First of all, we will not use topic modelling, but we will see the judgments as a bag-of-words, where pre-processing should be executed to decrease the number of features to improve efficiency and accuracy. Secondly, whereas Zheng retracted more than 2 million cases (also unpublished) about the entire field of law up to October 2016, we work on 15,000 published civil law cases until May 2017.

2.2 Tools

In this research, the following tools were used:

- KNIME: an open source platform focusing on data mining, manipulation, visualization and prediction. With its easy user interface, many machine learning applications can be used by building a workflow with different building blocks.[2] An example of the workflows built for this research can be seen below in **Figure 1**.
- MongoDB: an open source and free tool, focusing on storing data in JSON-like documents. It is possible to handle large amounts of data, to change data later on and to retrieve specific sets of data based on specific requirements. For example: retrieve all documents that are classified positively and contain 3 keywords. Each data point is stored as an object, where multiple instances can be added with different sizes.[3]

3. Classification Question 1

From rechtspraak.nl 15,000 cases were retrieved, starting with the newest from May 2017 and working 'down'. From these cases the XML was obtained, including all sorts of tags. For each case a new object was made in MongoDB with the XML as instance and the title as meta-data which had to be unique in the database. The tags were then stripped, the type of document was changed into txt-files, and these were added as new instances to each case in MongoDB. Next, these txt-instances were checked for the appearance of keywords referring to the Brussel I regulations. A legal expert, who also helped interpreting the results, made a list of words in multiple languages that indicate both the 44/2001 and 1215/2015 regulations. In Dutch the lists are as follows:

Brussel I: `Brussels I Regulation: EEX-Vo ; EG-Executieverordening; EEX-Verordening; Brussel I-Verordening; Brussel I; 44/2001`

[2] KNIME.COM AG. KNIME Open for Innovation. 2017. http://www.knime.org.
[3] MongoDB.COM AG. What is MongoDB. 2017. https://www.mongodb.com/.

Brussel I recast: `Brussels I Regulation recast: EEX-Vo II; Brussel Ibis; Brussel I-bis; EU-executieverordening; Brussel I bis-Verordening; EEXVerordening II; Brus-sel 1 bis-Vo; Brussel 1 bis; herschikte EEX-Vo; 1215/2012`

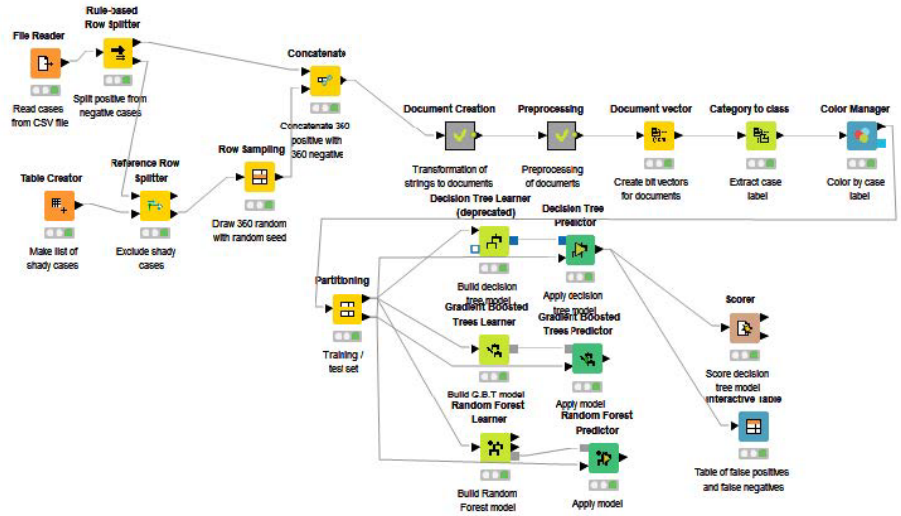

Figure 1: KNIME workflow used for the classification problems

The lists were combined and changed into one regular expression and a new instance 'clean' was created. In this instance the txt-file without the keywords was inserted. All the terms that matched the regular expression were listed as new instances and a new meta-data instance was set to 'true' if the case did contain a keyword, 'false' if it did not. See the example of **Figure 2** how all the instances are related. This way we created a labelled set of cases we can use for training and testing the classifiers.

It is important to understand that the labels 'true' and 'false' are used as 'golden standard'. However, this does not mean that there are no cases involving the regulation that are labelled false. Since we are researching the reliability of such a classification system, the incorrectly classified cases are important as well. As stated in the introduction, there are no agreements regarding referencing this regulation in case decisions, so the number of cases involving the regulation is expected to be larger than the number of 'true' cases.

Once this was done, an analysis of the true and false cases could be made. From the 15,000 cases, there were 360 cases classified as 'true', and 14,640 classified as 'false' using the keywords. Most positive cases contained between 1-3 keywords, but two even contained 6 keywords, namely ECLI:NL:GHSHE:2017:1873 and ECLI:NL:GHSHE:2017:1874 and these two are related. The frequency distribution of the keywords over the documents can be seen in **Table 1**.

From all these instances, a CSV-file of 15,000 rows was created with the columns 'title', 'document' and 'label' containing the title of the case, the txt-file of judgment without keywords and the classification true or false respectively. This CSV-file was then ready to be handled by KNIME.

Figure 2: Example of positive case in MongoDB with all instances.

Table 1: The number of documents each keyword appears in.

Keyword (in Dutch)	Frequency
1215/2012	103
44/2001	72
Brussel I	134
Brussel I bis	-
Brussel I Bis-Verordening[4]	8
Brussel I bis-Vo	-
Brussel Ibis	12
Brussel I-bis	3
Brussel I-Verordening	7
EEX-Verordening	123
EEX-Verordening II	7
EEX-Vo	71
EEX-Vo II	-
EG-Executieverordening	1
EU-Executieverordening	1
Herschikte EEX-Vo	23

3.1 Pre-processing of data

To be able to use algorithms on the data, a binary vector of unique terms for each case was needed. However, since the texts of the cases were sometimes very extensive and the number of positive examples scarce, proper pre-processing was important:

1. For each document delete:
    - Terms consisting of the characters: !#$%()*+,./:;¡¿=?@ˆ ‘— []
    - Terms consisting only of numbers
    - Terms consisting of less than 4 characters
    - Terms that occur in a stop list or occur in more than 95% of the documents

[4] ‘Verordening’ = Regulation; ‘executieverordening’ = implementing regulation.

- Terms that occur in less than 1% of the documents

2. For each term in each document:
   - Convert all characters to lowercase
   - Use Snowball Stemmer for Dutch language[5]

This resulted in around 6,000 unique terms that were then used as features. Of these 6,000 terms a few stood out, because of their similarity with the keywords. These terms were further investigated and also analysed by our domain expert. He concluded that most of these keywords have no relation to Brussel I, so to be certain not to train the model on wrongly labelled cases, the 34 cases containing one of these 'grey keywords' were excluded from the total set before selecting the test- and training data.

3.2 Experimental Setup

To create a baseline of 50%, 360 negative examples were drawn randomly next to the 360 positive examples. To enlarge reliability of this random sample, this was done with 10 different random seeds. Each experiment of 720 cases was then split in 504 cases to train on (70%) and 216 cases to test the classification on (30%).

Since earlier experiments already showed the poor results of the algorithms naive Bayes (accuracy of 0.51) and k-nearest neighbour (accuracy of 0.77), these were excluded in further experiments. We noted that algorithms based on trees resulted in the best accuracy, so we will use decision trees, gradient boosting trees and random forest.

Decision trees is a tree-structured algorithm, where each internal node presents a test on an attribute, each branch corresponds to an attribute value and each leaf node represents a class label. Decision trees can deal with noisy data and function well with disjunctive hypotheses [7]. It does not have any requirements about the distribution of the data (for example Naive Bayes requires independent variables), since it is a non-parametric technique.

Gradient Boosting trees is an algorithm that keeps improving its model by calculating the error and fitting new Decision Trees to the corresponding cost function, and by doing so increasing its complexity. Lawrence e.a. [7] state that in most cases it outperforms decision trees or at least performs equally. It is said to deal with overfitting better than decision trees.

Random forest is a machine learning algorithm that again uses decision trees, by learning multiple decision trees simultaneously. It then chooses the most common label of all the models. This has the advantage of decreasing the overfitting problem that decision trees tend to have. But Prasad e.a. [8] state the disadvantages of time and computational resources and the 'black-box' characteristic.

In the pre-processing all the cases were labelled true or false based on the occurrence of a few keywords. The research question is to what extent the classifier can classify cases based on their texts without these keywords. This can be measured by using the relative number of correctly classified cases, also known as the accuracy. Precision is the number of correctly classified positive examples divided by the number of examples labelled by the system as positive and recall is the number of correctly classified positive examples divided by the number of positive examples in the data. The F1-measure is the harmonic mean of both precision and recall. Since this study is mainly interested in cases that involve the regulation (positive cases), but are not

[5] http://snowball.tartarus.org/algorithms/dutch/stemmer.html

classified as true because of a lack of keywords or lack of agreement on referencing, recall is in this case more important than precision. A measure that weighs recall higher than precision is the F2-measure and for each experiment this value was calculated, its formula is as follows:

$$F_2 = 5 * \frac{precision * recall}{4 * precision + recall}$$

3.3 Results

Table 2 presents an overview of the various measures for the average over 10 samples for the three methods. In each sample we drew 360 negative cases together with all 360 positive ones. Random forest scores best on all performance measures. All methods do much better than the baseline of 50%.

Table 2: Overview of average of 10 samples of multiple evaluation methods using different algorithms. + means case involving the regulation and - means not.

Sample	Recall-	Recall+	Precision-	Precision+	F1-	F1+	F2-	F2+	Accuracy
Decision trees	0.91	0.90	0.91	0.91	0.91	0.90	0.85	0.88	0.91
Gradient boosting	0.90	0.90	0.91	0.90	0.90	0.90	0.90	0.90	0.91
Random forest	0.94	0.95	0.95	0.94	0,94	0.95	0.95	0.94	0.95

4. Classification Question 2

For the second classifier the goal was to reliably distinguish old cases from new cases. So all the negative examples were excluded and only the 360 positive examples were used. The keywords that were used for the first classifier, were used again, but this time they were not combined. New instances were then created in MongoDB, indicating whether cases were labelled 'old' or 'new' (or both). The labelling resulted in 310 new cases, 124 old cases, of which 74 were labelled both new and old. In Table 2 the frequency of each keyword can be found for both old and new cases.

Table 3: The number of documents each keyword appears in, divided in old cases (44/2001) and new cases (1215/2015) where overlap is possible.

Keyword	Frequency 'old'	Frequency 'new'
1215/2012	65	103
44/2001	72	18
Brussel I	134	18
Brussel I bis	-	-
Brussel I Bis-Verordening[6]	2	8
Brussel I bis-Vo	-	-
Brussel Ibis	6	12

[6] 'Verordening' = Regulation; 'executieverordening' = implementing regulation.

Brussel I-bis	2	3
Brussel I-Verordening	7	1
EEX-Verordening	123	37
EEX-Verordening II	2	7
EEX-Vo	71	29
EEX-Vo II	-	-
EG-Executieverordening	1	-
EU-Executieverordening	1	1
Herschikte EEX-Vo	12	23

4.1 Experimental Setup

The pre-processing steps were the same as with the first experiment. This classification problem was split into two parts: old against not old and new against not new. By doing so, the classification remained binary, just like in the first experiment. In the first classification problem there were 50% positive examples and 50% negative examples and the baseline could be set to 50. In this second classification problem that is different. For the old cases the baseline is 310/360 = 86% and for the new cases (360-124)/360 = 66%.

Again, the data was split into a train and test set using a 70/30 ratio. Because of the small number of new cases, we also used a 80/20 ratio for that part. The same machine learning algorithms were used as in the first classifier: decision trees, gradient boosting trees and random forest. For the algorithm gradient boosting trees, this time also multiple settings were performed for the number of maximum tree-depth. In the first classifier this hardly changed the accuracy, but now it did, as can be seen below.

4.2 Results

From **Table 4** below it can be concluded that the baseline of 86% for classifying old cases was not reached by any machine learning algorithm. The highest accuracy was obtained by using gradient boosting trees with a threshold of maximum tree-depth set on 4 (0.85). Also when looking at the F1-values this algorithm outperforms the rest.

Table 4: Results of classifying old cases (+) against not old cases (-).

Sample	Recall-	Recall+	Precision-	Precision+	F1-	F1+	F2-	F2+	Accuracy
Decision trees	0.27	0.98	0.75	0.84	0.40	0.90	0.31	0.95	0.84
Gradient boosting max 10	0.36	0.90	0.47	0.84	0.41	0.87	0.38	0.89	0.79
Gradient boosting max 4	0.27	1	1	0.85	0.43	0.92	0.32	0.97	0.85
Random forest	0.91	0.98	0.50	0.81	0.15	0.89	0.11	0.94	0.80

From **Table 5** and **Table 6** it can be concluded that for classifying new cases, the baseline of 66% is met by all algorithms. According to the highest accuracy, again gradient boosting trees with a threshold of maximum tree-depth set to 4 was best. Only for random forest the accuracy was lower when using 80/20 ratio instead of 70/30.

Although random forest (80/20) obtained the highest re-call for the negative examples, it got the lowest recall for the positive examples. When looking at the F1-values (80/20) it can be seen that gradient boosting with threshold 4 scores best on both positive and negative examples. This also holds for the F2-values and as stated before, for the accuracy.

Table 5: Results of classifying new cases (+) against not new cases (-) with 70/30 ratio for train-test set.

Sample	Recall-	Recall+	Precision-	Precision+	F1-	F1+	F2-	F2+	Accuracy
Decision trees	0.83	0.56	0.73	0.69	0.77	0.62	0.58	0.81	0.72
Gradient boosting max 10	0.80	0.62	0.75	0.68	0.77	0.65	0.63	0.79	0.73
Gradient boosting max 4	0.91	0.58	0.75	0.81	0.82	0.68	0.61	0.87	0.77
Random forest	0.91	0.44	0.70	0.77	0.79	0.57	0.49	0.86	0.72

Table 6: Results of classifying new cases (+) against not new cases (-) with 80/20 ratio for train-test set.

Sample	Recall-	Recall+	Precision-	Precision+	F1-	F1+	F2-	F2+	Accuracy
Decision trees	0.79	0.67	0.77	0.69	0.78	0.68	0.67	0.78	0.74
Gradient boosting max 4	0.86	0.67	0.78	0.77	0.82	0.71	0.69	0.84	0.78
Random forest	0.88	0.40	0.67	0.70	0.76	0.51	0.44	0.83	0.68

5. Conclusions and Discussion

The objective of the research was to answer the following question: To what extent is it possible to design a supervised classification system that uses judgments of civil law cases from rechtspraak.nl to distinguish:

1. cases about Brussels I Regulation from all the other civil law cases?
2. cases from the Brussels I Regulation Recast from the Brussels I Regulation?

Classifying Brussels I Regulation cases from other civil law cases can be done with an accuracy of 0.95, but with computational limitations. Classifying Brussels I Regulation from the Brussels I Regulation Recast is harder, due to the small amount of data.

With only 124 cases referring to the 1215/2015 regulation, the second classifier did not perform well on accuracy. The number of false positives and false negatives was high and further research is needed to find out what these incorrectly classified cases say about the data. Are they counter examples due to too few data or was the initial labelling incorrect?

Because of limitations in computer power, we decided to use only 360 negative cases in the first classifier. However, choosing 360 out of 14,640 can be done in many ways and even though this was done 10 different times this may have influenced the results.

To draw conclusions for the entire European Union, it is necessary to expend the research to other languages. We have keywords for several languages, but we may need access to legal experts from these jurisdictions to help analysing results. Extending research to other countries will also increase the number of positive cases.

Another option is using unpublished cases from courts, but we will have to see whether their format is similar to the cases from the official portal.

The performance of the classification systems was evaluated against the original labelling of the cases based on keyword matching. If the original labelling was not correct, this will of course influence the performance evaluation. We did some analysis of the original labelling and removed some cases which contained 'grey area' keywords, but a thorough analysis by human experts would be better. However, given the nature of the work and the scarcity of experts, this was not feasible at present. Our approach has the benefit of little human effort, but the disadvantage of possible mistakes in classification. Since the final analysis of the impact of legal change will be done by human experts anyway, we do not see this as a major problem.

Acknowledgements. Part of this research is co-funded by the Civil Justice Programme of the European Union in the Brussel I project under grant JUST/2014/JCOO/AG/CIVI/7754. We would like to thank our domain expert Michiel van Rooijen.

6. References

[1] Danov, M. "The Brussels I Regulation: Cross-Border Collective Redress Proceedings and Judgments". In: *Journal of Private International Law* 6.2 (2017), pp. 359–393.

[2] Bruninghaus, S. and Ashley, K.D. "Toward Adding Knowledge to Learning Algorithms for Indexing Legal Cases". In: *Proceedings of the 7th international conference on Artificial intelligence and law*. ACM, 1999, pp. 9–17.

[3] Maat, E. de, Krabben, K. and Winkels, R. "Machine Learning versus Knowledge Based Classification of Legal Texts". In: JURIX (2010), IOS press, pp. 87–96.

[4] Goncalves, T. and Quaresma, P. "Is linguistic information relevant for the classification of legal texts?" In: *Proceedings of the 10th international conference on Artificial intelligence and law*. ACM, 2005, pp. 168–176.

[5] Forman, G. "An extensive empirical study of feature selection metrics for text classification". In: *Journal of machine learning research* 3.Mar (2003), pp. 1289–1305.

[6] Zheng, K.H.. *Brussel I project*. Tech. rep. University of Amsterdam, 2016.

[7] Lawrence, R., Bunn, A., Powell, S. & Zambon, M. "Classification of remotely sensed imagery using stochastic gradient boosting as a refinement of classification tree analysis". In: *Remote sensing of environment 90:3* (2004), pp. 331–336.

[8] Prasad, A.M., Iverson, L.R., and Liaw, A. "Newer classification and regression tree techniques: bagging and random forests for ecological prediction". In: *Ecosystems 9:2* (2006), pp. 181–199.