Text Mining of Supreme Administrative Court Jurisdictions

Ingo Feinerer and Kurt Hornik

Department of Statistics and Mathematics,
Wirtschaftsuniversität Wien, A-1090 Wien, Austria
{h0125130, Kurt.Hornik}@wu-wien.ac.at

Abstract. Within the last decade text mining, i.e., extracting sensitive information from text corpora, has become a major factor in business intelligence. The automated textual analysis of law corpora is highly valuable because of its impact on a company's legal options and the raw amount of available jurisdiction. The study of supreme court jurisdiction and international law corpora is equally important due to its effects on business sectors.

In this paper we use text mining methods to investigate Austrian supreme administrative court jurisdictions concerning dues and taxes. We analyze the law corpora using R with the new text mining package **tm**. Applications include clustering the jurisdiction documents into groups modeling tax classes (like income or value-added tax) and identifying jurisdiction properties. The findings are compared to results obtained by law experts.

1 Introduction

A thorough discussion and investigation of existing jurisdictions is a fundamental activity of law experts since convictions provide insight into the interpretation of legal statutes by supreme courts. On the other hand, text mining has become an effective tool for analyzing text documents in automated ways. Conceptually, clustering and classification of jurisdictions as well as identifying patterns in law corpora are of key interest since they aid law experts in their analyses. E.g., clustering of primary and secondary law documents as well as actual law firm data has been investigated by Conrad et al. (2005). Schweighofer (1999) has conducted research on automatic text analysis of international law.

In this paper we use text mining methods to investigate Austrian supreme administrative court jurisdictions concerning dues and taxes. The data is described in Section 2 and analyzed in Section 3. Results of applying clustering and classification techniques are compared to those found by tax law experts. We also propose a method for automatic feature extraction (e.g., of the senate size) from Austrian supreme court jurisdictions. Section 4 concludes.

2 Administrative Supreme Court jurisdictions

2.1 Data

The data set for our text mining investigations consists of 994 text documents. Each document contains a jurisdiction of the Austrian supreme administrative court (Verwaltungsgerichtshof, VwGH) in German language. Documents were obtained through the legal information system (Rechtsinformationssystem, RIS; `http://ris.bka.gv.at/`) coordinated by the Austrian Federal Chancellery. Unfortunately, documents delivered through the RIS interface are HTML documents oriented for browser viewing and possess no explicit metadata describing additional jurisdiction details (e.g., the senate with its judges or the date of decision). The data set corresponds to a subset of about 1000 documents of material used for the research project "Analyse der abgabenrechtlichen Rechtsprechung des Verwaltungsgerichtshofes" supported by a grant from the Jubiläumsfonds of the Austrian National Bank (Oesterreichische Nationalbank, OeNB), see Nagel and Mamut (2006). Based on the work of Achatz et al. (1987) who analyzed tax law jurisdictions in the 1980s this project investigates whether and how results and trends found by Achatz et al. compare to jurisdictions between 2000 and 2004, giving insight into legal norm changes and their effects and unveiling information on the quality of executive and juristic authorities. In the course of the project, jurisdictions especially related to dues (e.g., on a federal or communal level) and taxes (e.g., income, value-added or corporate taxes) were classified by human tax law experts. These classifications will be employed for validating the results of our text mining analyses.

2.2 Data preparation

We use the open source software environment R for statistical computing and graphics, in combination with the R text mining package **tm** to conduct our text mining experiments. R provides premier methods for clustering and classification whereas **tm** provides a sophisticated framework for text mining applications, offering functionality for managing text documents, abstracting the process of document manipulation and easing the usage of heterogeneous text formats.

Technically, the jurisdiction documents in HTML format were downloaded through the RIS interface. To work with this inhomogeneous set of malformed HTML documents, HTML tags and unnecessary white space were removed resulting in plain text documents. We wrote a custom parsing function to handle the automatic import into **tm**'s infrastructure and extract basic document metadata (like the file number).

3 Investigations

3.1 Grouping the jurisdiction documents into tax classes

When working with larger collections of documents it is useful to group these into clusters in order to provide homogeneous document sets for further investigation by

experts specialized on relevant topics. Thus, we investigate different methods known in the text mining literature and compare their results with the results found by law experts.

k-means Clustering

We start with the well known k-means clustering method on term-document matrices. Let $\mathrm{tf}_{t,d}$ be the frequency of term t in document d, m the number of documents, and df_t is the number of documents containing the term t. Term-document matrices M with respective entries $\omega_{t,d}$ are obtained by suitably weighting the term-document frequencies. The most popular weighting schemes are *Term Frequency* (*tf*), where $\omega_{t,d} = \mathrm{tf}_{t,d}$, and *Term Frequency Inverse Document Frequency* (*tf-idf*), with $\omega_{t,d} = \mathrm{tf}_{t,d} \log_2(m/\mathrm{df}_t)$, which reduces the impact of irrelevant terms and highlights discriminative ones by normalizing each matrix element under consideration of the number of all documents. We use both weightings in our tests. In addition, text corpora were stemmed before computing term-document matrices via the **Rstem** (Temple Lang, 2006) and **Snowball** (Hornik, 2007) R packages which provide the Snowball stemming (Porter, 1980) algorithm.

Domain experts typically suggest a basic partition of the documents into three classes (income tax, value-added tax, and other dues). Thus, we investigated the extent to which this partition is obtained by automatic classification. We used our data set of about 1000 documents and performed k-means clustering, for $k \in \{2, \ldots, 10\}$. The best results were in the range between $k = 3$ and $k = 6$ when considering the improvement of the within-cluster sum of squares. These results are shown in Table 1. For each k, we compute the agreement between the k-means results based on the term-document matrices with either *tf* or *tf-idf* weighting and the expert rating into the basic classes, using both the Rand index (Rand) and the Rand index corrected for agreement by chance (cRand). Row "Average" shows the average agreement over the four ks. Results are almost identical for the two weightings employed. Agreements are rather low, indicating that the "basic structure" can not easily be captured by straightforward term-document frequency classification.

Table 1. Rand index and Rand index corrected for agreement by chance of the contingency tables between k-means results, for $k \in \{3,4,5,6\}$, and expert ratings for *tf* and *tf-idf* weightings.

	Rand		cRand	
k	*tf*	*tf-idf*	*tf*	*tf-idf*
3	0.48	0.49	0.03	0.03
4	0.51	0.52	0.03	0.03
5	0.54	0.53	0.02	0.02
6	0.55	0.56	0.02	0.03
Average	0.52	0.52	0.02	0.03

We note that clustering of collections of large documents like law corpora presents formidable computational challenges due to the dimensionality of the term-document

matrices involved: even after stopword removal and stemming, our about 1000 documents contained about 36000 different terms, resulting in (very sparse) matrices with about 36 million entries. Computations took only a few minutes in our cases. Larger datasets as found in law firms will require specialised procedures for clustering high-dimensional data.

Keyword based Clustering

Based on the special content of our jurisdiction dataset and the results from *k*-means clustering we developed a clustering method which we call *keyword based clustering*. It is inspired by simulating the behaviour of tax law students preprocessing the documents for law experts. Typically the preprocessors skim over the text looking for discriminative terms (i.e., keywords). Basically, our method works in the same way: we have set up specific keywords describing each cluster (e.g., "income" or "income tax" for the income tax cluster) and analyse each document on the similarity with the set of keywords.

Fig. 1. Plot of the contingency table between the keyword based clustering results and the expert rating.

Figure 1 shows a mosaic plot for the contingency table of cross-classifications of keyword based clustering and expert ratings. The size of the diagonal cells (visualizing the proportion of concordant classifications) indicates that the keyword based clustering methods works considerably better than the *k*-means approaches, with a

Rand index of 0.66 and a corrected Rand index of 0.32. In particular, the expert "income tax" class is recovered perfectly.

3.2 Classification of jurisdictions according to federal fiscal code regulations

A further rewarding task for automated processing is the classification of jurisdictions into documents dealing and into documents not dealing with Austrian federal fiscal code regulations (Bundesabgabenordnung, BAO).

Due to the promising results obtained with string kernels in text classification and text clustering (Lodhi et al., 2002; Karatzoglou and Feinerer, 2007) we performed a "C-svc" classification with support vector machines using a full string kernel, i.e., using

$$k(x,y) = \sum_{s \in \Sigma^*} \lambda_s \cdot \nu_s(x) \cdot \nu_s(y)$$

as the kernel function $k(x,y)$ for two character sequences x and y. We set the decay factor $\lambda_s = 0$ for all strings $|s| > n$, where n denotes the document lengths, to instantiate a so-called full string kernel (full string kernels are computationally much better natured). The symbol Σ^* is the set of all strings (under the Kleene closure), and $\nu_s(x)$ denotes the number of occurrences of s in x.

For this task we used the **kernlab** (Karatzoglou et al., 2006; Karatzoglou et al., 2004) R package which supports string kernels and SVM enabled classification methods. We used the first 200 documents of our data set as training set and the next 50 documents as test set. We compared the 50 received classifications with the expert ratings which indicate whether a document deals with the BAO by constructing a contingency table (confusion matrix). We received a Rand index of 0.49. After correcting for agreement by chance the Rand index floats around at 0. We measured a very long running time (almost one day for the training of the SVM, and about 15 minutes prediction time per document on a 2.6 GHz machine with 2 GByte RAM).

Therefore we decided to use the classical term-document matrix approach in addition to string kernels. We performed the same set of tests with *tf* and *tf-idf* weighting, where we used the first 200 rows (i.e, entries in the matrix representing documents) as training set, the next 50 rows as test set.

Table 2. Rand index and Rand index corrected for agreement by chance of the contingency tables between SVM classification results and expert ratings for documents under federal fiscal code regulations.

	tf	*tf-idf*
Rand	0.59	0.61
cRand	0.18	0.21

Table 2 presents the results for classifications obtained with both *tf* and *tf-idf* weightings. We see that the results are far better than the results obtained by employing string kernels.

These results are very promising, and indicate the great potential of employing support vector machines for the classification of text documents obtained from jurisdictions in case term-document matrices are employed for representing the text documents.

3.3 Deriving the senate size

Table 3. Number of jurisdictions ordered by senate size obtained by fully automated text mining heuristics. The percentage is compared to the percentage identified by humans.

Senate size	0	3	5	9
Documents	0	255	739	0
Percentage	0.000	25.654	74.346	0.000
Human Percentage	2.116	27.306	70.551	0.027

Jurisdictions of the Austrian supreme administrative court are obtained in so-called senates which can have 3, 5, or 9 members, with size indicative of the "difficulty" of the legal case to be decided. (It is also possible that no senate is formed.) An automated derivation of the senate size from jurisdiction documents would be highly useful, as it would allow to identify structural patterns both over time and across areas. Although the formulations describing the senate members are quite standardized it is rather hard and time-consuming for a human to extract the senate size from hundreds of documents because a human must read the text thoroughly to differ between senate members and auxiliary personnel (e.g., a recording clerk). Thus, a fully automated extraction would be very useful.

Since most documents contain standardized phrases regarding senate members (e.g., "The administrative court represented by president Dr. X and the judges Dr. Y and Dr. Z ... decided ...") we developed an extraction heuristic based on widely used phrases in the documents to extract the senate members. In detail, we investigate punctuation marks and copula phrases to derive the senate size. Table 3 summarizes the results for our data set by giving the total number of documents for senate sizes of zero (i.e., documents where no senate was formed, e.g., due to dismissal for want of form), three, five, or nine members. The table also shows the percentages and compares these to the aggregated percentages of the full data set, i.e., $n > 1000$, found by humans. Figure 2 visualizes the results from the contingency table between machine and human results in form of an agreement plot, where the observed and expected diagonal elements are represented by superposed black and white rectangles, respectively. The plot indicates that the extraction heuristic works very well. This is supported by the very high Rand index of 0.94 and by the corrected Rand index of 0.86.

Further improvements could be achieved by saving identified names of judges in order to identify them again in other documents. Of course, ideally information such as senate size would be provided as metadata by the legal information system, per-

haps even determined automatically by text mining methods for "most" documents (with a per-document measure of the need for verification by humans).

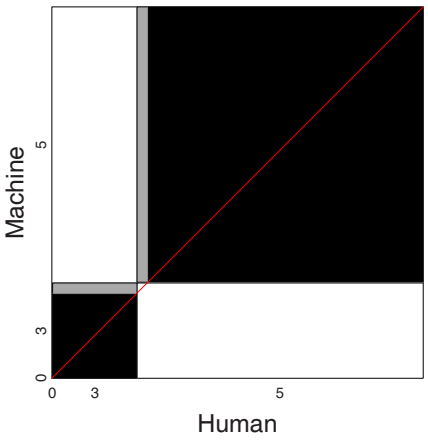

Fig. 2. Agreement plot of the contingency table between the senate size reported by text mining heuristics and the senate size reported by humans.

4 Conclusion

In this paper we have presented approaches to use text mining methods on (supreme administrative) court jurisdictions. We performed k-means clustering and introduced keyword based clustering which works well for text corpora with well defined formulations as found in tax law related jurisdictions. We saw that the clustering works well enough to be used as a reasonable grouping for further investigation by law experts. Second, we investigated the classification of documents according to their relation to federal fiscal code regulations. We used both string kernels and term-document matrices with *tf* and *tf-idf* weighting as input for support vector machine based classification techniques. The experiments unveiled that employing term-document matrices yields both superior performance as well as fast running time. Finally, we considered a situation typical in working with specialized text corpora, i.e., we were looking for a specific property in each text corpus. In detail we derived the senate size of each jurisdiction by analyzing relevant text phrases considering punctuation marks, copulas and regular expressions. Our results show that text mining methods can clearly aid legal experts to process and analyze their law document corpora, offering both considerable savings in time and cost as well as the possibility to conduct investigations barely possible without the availability of these methods.

Acknowledgments

We would like to thank Herbert Nagel for providing us with information and giving us feedback.

References

ACHATZ, M., KAMPER, K., and RUPPE H. (1987): Die Rechtssprechung des VwGH in Abgabensachen. Orac Verlag, Wien.

CONRAD, J., AL-KOFAHI, K., ZHAO, Y. and KARYPIS, G. (2005): Effective Document Clustering for Large Heterogeneous Law Firm Collections. In: *10th International Conference on Artificial Intelligence and Law (ICAIL)*. 177–187.

FEINERER, I. (2007): **tm**: Text Mining Package, R package version 0.1-2.

HORNIK, K. (2007): **Snowball**: Snowball Stemmers, R package version 0.0-1.

KARATZOGLOU, A. and FEINERER, I. (2007): Text Clustering with String Kernels in R. In: *Advances in Data Analysis (Proceedings of the 30th Annual Conference of the GfKl)*. 91–98. Springer-Verlag.

KARATZOGLOU, A., SMOLA, A. and HORNIK, K. (2006): **kernlab**: Kernel-based machine learning methods including support vector machines, R package version 0.9-1.

KARATZOGLOU, A., SMOLA, A., HORNIK, K. and ZEILEIS, A. (2004): **kernlab** — An S4 Package for Kernel Methods in R. *Journal of Statistical Software, 11(9), 1–20*.

LODHI, H., SAUNDERS, C., SHAWE-TAYLOR, J., WATKINS, C., and CRISTIANINI, N. (2002): Text classification using string kernels. *Journal of Machine Learning Research, 2, 419–444*.

NAGEL, H. and MAMUT, M. (2006): Rechtsprechung des VwGH in Abgabensachen 2000–2004.

PORTER, M. (1980): An algorithm for suffix stripping. *Program, 14(3), 130–137*.

R DEVELOPMENT CORE TEAM (2006): R*: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org/`.

SCHWEIGHOFER, E. (1999): *Legal Knowledge Representation, Automatic Text Analysis in Public International and European Law*. Kluwer Law International, Law and Electronic Commerce, Volume 7, The Hague. ISBN 9041111484.

TEMPLE LANG, D. (2006): **Rstem**: Interface to Snowball implementation of Porter's word stemming algorithm, R package version 0.3-1.