Towards legal change analysis: clustering of Polish Civil Code amendments

Łukasz Górski
Interdisciplinary Centre for Mathematical and Computational Modelling
University of Warsaw
lgorski@icm.edu.pl

ABSTRACT

Due to the growing activity of legislators, lawyers are in need of tools that would allow them to get a better understanding of an ever-growing corpus of legislative materials. Herein we propose a tool that visualizes and clusters thematically similar amending acts, allowing a lawyer to quickly review related provisions, thus giving an insight into a legislative history of a given legal institution. The methods suggested herein (based on TF-IDF, word and paragraph embeddings and PCA as well as k-mean clustering) are evaluated on the provisions of the Polish Civil Code.

1 INTRODUCTION

This paper describes first steps undertaken in the development of a software solution used for the visualization of legal change, which aims to provide the user with means to effectively explore a database of amending acts. We aim to develop a solution which is able to group together amending acts that are thematically similar, in an unsupervised manner.

The proof-of-concept implementation studied herein has been tested using the Polish Civil Code and relevant amending acts issued from its enactment in 1965 up to November 2018. This legal act was chosen as a basis for experiments for the following reasons:

(i) While the Code was in force, the Polish economy has undergone transformation from socialism to capitalism and later its law had be adapted to the law of the European Union. The processes pertaining to the recognition of the information and communication technologies in the domain of law were also reflected in the Code. Turbulent times, in which the Code existed, made it subject to almost 90 amending acts. Some of the sections composing the Civil Code were, in fact, subject to change multiple times - please consult the heat map (Fig. 1) for a graphical representation of the number of times a given legal section was amended. Therefore this research aimed to assess whether modern machine learning approaches would be able to recognize and discover discrete categories of changes (not necessarily the three mentioned hereinbefore), based only on the text of relevant legal provisions.

(ii) Even though the amending acts should be as straight-forward to understand and as precise as possible, the legislative practice does not always live up to this standard. For example, the titles of the amending acts do not help in the clustering task, as those are often called simply "Statute amending Civil Code" (and sometimes statutes amending the Code focus on different pieces of legislation at the same time and the Code may not be mentioned in their title at all) or the amending provisions can be scattered in a number of statutes that hold other substantive provisions.

(iii) While there has been an extensive body of research pertaining to the use of machine learning in the area of law in the English language, the body of research pertaining to Polish law is obviously smaller.

In: Proceedings of the Third Workshop on Automated Semantic Analysis of Information in Legal Text (ASAIL 2019), June 21, 2019, Montreal, QC, Canada.

2 RELATED WORK

Practising lawyers need tools that would allow them to track legal changes, especially due to the increasing activity of legislatures. For example, in the Polish legal system it has been noted a number of times that currently the legal system is undergoing the process of "inflation of law". This notion was recognized by theorists [16] and even the courts, one of which explicitly stated that the legislature is currently *multiplying the numbers of unnecessary statutes, which makes accessing ... sources of law difficult* [6].

The problem of orientation in a dynamically changing system of statues can be mitigated to a degree by the introduction of consolidated texts of acts. In practice, in Poland, the process of consolidation of legal texts is two-fold. On the one hand, there are official consolidated texts of legal acts published by the authorities. In practice, those are however seldom used. Lawyers routinely use the legal databases and search engines that are developed by private companies (legal information systems) instead. Currently, the market remains split between C.H. Beck, developer of Legalis information system, and Wolters Kluwer Polska, with their Lex system. The editorial offices of both of these systems carefully analyse every amending act and issue their versions of the consolidated text. Obviously, the consolidated texts published by those privately-owned enterprises do not have a formal force of law, yet the convenience offered by them makes those closed and paid platforms a go-to solution for professionals. As far as the recognition of amendments goes, both of these systems offer, *inter alia*, a clear diff-like view of the legislative history of a given legal provision (Fig. 2).

However, those solutions do not employ any form of graphical presentation of amendments. In fact, artificial intelligence methods are used sparsely in those types of software: for example the consolidated versions of statutes are created mainly by hand [7]. Therefore this research, independent of aforementioned commercial solutions, aims to look into means of extending already existing systems.

As far as the analysis of amending acts in the AI and Law community goes, the focus up to this time was mainly on the automatic

Figure 1: Heatmap showing the number of times each section of the Polish Civil Code was amended. All sections were sorted sequentially by their numbers. Inspired by the traditional division put forth by the 19th-century German school of Pandectists, the Code is divided into four books - the starting articles for those books were marked for reference.

2016.10.07 Dz.U. z 2016 r. poz. 1579 zm.

Art. 78 ^{1} [Elektroniczna forma czynności prawnej]

§ 1. Do zachowania elektronicznej formy czynności prawnej wystarcza złożenie oświadczenia woli w postaci elektronicznej i opatrzenie go kwalifikowanym ~~bezpiecznym~~ podpisem elektronicznym ~~weryfikowanym przy pomocy ważnego kwalifikowanego certyfikatu~~.
§ 2. Oświadczenie woli złożone w formie elektronicznej jest równoważne z oświadczeniem woli złożonym w formie pisemnej ~~, chyba że ustawa lub czynność prawna zastrzega inaczej~~.

Figure 2: Diff view of amended statute in Legalis system. Additions are in blue and underlined, while deletions are denoted by red and crossed out. Competing Lex system offers a similar view. See the Table 1 for the English translation of the passage.

consolidation of legal texts. For example, authors in [15] created a tool for semiautomatic implementation of amending acts. Similar subject was undertaken in [1], in which a feasibility of using an SGML-based engine for amendments processing was explored. Dually, in [2] a drafting environment was prototyped, which generated amending acts based on amendments introduced by drafter into a principal act.

Whilst this research uses word embeddings techniques as well as older TF-IDF-based methods, the feasibility of using word embeddings in eDiscovery procedures was in fact already explored. In [18] a *Disco* system is described, which uses word2vec word embeddings to help legal expert with refining her document database search queries. In Poland, doc2vec model was already used in SAOS, a Polish courts' judgment analysis system, as a basis for similarity analysis module [4]. [8] focused on the explainability of AI methods and supplemented text similarity measures (based on TFIDF and word embeddings) with metric showing how much each word contributes to overall similarity result when comparing text phrases. K-Means clustering employed in this research was used with, *inter alia*, embedding-based methods for grouping controversial issues that were extracted from Chinese legal texts [17]. Similarly, other authors clustered the documents regarding Chinese criminal cases [5].

3 METHODS

In pursuit of the aim outlined in the preceding section a pipeline of an existing tools has been created, with all of them instrumented by Python programms. Python 3.6.8 from Anaconda was used for text processing instrumentation, as well as: gensim 3.4.0 for TF-IDF and embeddings calculations, scikit-learn 0.20.2 for clustering, pandas 0.24.0 for data manipulation, nltk 3.4 for text processing and matplotlib 3.0.2 for visualization.

Text processing pipeline can be divided into the following phases:

- The **generation** phase involves reading the consolidated versions of a given statute and extracting the differences between each successive version. In this phase a textual representation of changes, similar to that shown in Fig. 2, is created. The amending acts are not directly processed: this problem, while itself interesting, is out of the scope of this paper. Usable diffs can be created using Linux `wdiff` command. In fact, for the purpose of this study, a number of diff-generating tools were tested, yet `wdiff` seemed to be best suited for our instant needs, offering the clearest results (Table 1 can be consulted for examples of differences between the output of various diff-generating tools).

The extraction of diffs allowed the creation of three different bodies of amendments corpora. For their detailed description and example Table 2 should be consulted. The first corpus version (C_1) consisted of a complete text of given legal sections after amending; the second version (C_2) included only the words that were inserted into a given legal section. However, both of these corpora did not include the texts that were deleted by an amending act. Yet, the provisions or parts of them that were struck down can carry at least the same amount of semantic meaning as those that were left untouched or added by the legislature. Moreover, in contemporary legal systems, legislative action is not the only means of changing the statute. For example, in Poland, the Constitutional Tribunal was called a "negative legislator". This term means that, in principle, a Tribunal is unable to amend a given legal act by adding some provisions, yet is perfectly capable of striking a given provision down. While this position is overly simplistic (as Tribunal in practice was able to pass, *inter alia*, interpretative judgments, in which it concludes that a given provision is in accordance with the Constitution as long as its interpretation is in line with the one put forth by the Tribunal [19]) we should be able to include in our clusterization endeavour effects of removal of a given statutory provision. To achieve this aim, for the purpose of this study, a third version of the corpus (C_3) included the parts of the legal provisions that were inserted by the amending acts alongside the deleted ones. The disadvantage of this technique is that it distorts the natural flow of the text

Table 1: Differences old and new versions of a given legal provision (section 78[1] § 1 of the Civil Code, as amended by the amending act published in Dz.U. [Journal of Laws] from 2016, item 1579), as shown by different implementations of diff (for illustrative purposes the English translation of original Polish passage is used[1]).

Diff command used	Result
`wdiff`	§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a ~~secure electronic signature verified with a valid~~ qualified ~~certificate.~~ electronic signature.
`diff`	~~§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a secure electronic signature verified with a valid qualified certificate.~~ § 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a qualified electronic signature.
`difflib` (Python library)	§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a ~~se~~equ~~r~~alifiedelectronic signatur~~e verified with a valid qualified certificat~~e.
`simplediff` (Python library)	§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a ~~se~~equ~~r~~alifiedelectronic signatur~~e verified with a valid qualified certificat~~e.

Table 2: Corpus types for further down the line processing.

Symbol	Description	Example from the Civil Code
C_1	Legal section's text after amendments	§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a qualified electronic signature.
C_2	Only words inserted by the amending act	electronic signature
C_3	Using crossed out parts of a given section alongside the inserted ones	§ 1. In order to observe the electronic form of an act in law it shall be sufficient to make a declaration of intent in electronic form and provide it with a secure electronic signature verified with a valid qualified certificate electronic signature.

and might not fare well with a paragraph embedding method that depends on the natural sequence of words in a sentence, and might be better suited for methods that employ bag of words technique.

- In **preprocessing** phase these three variations of corpora were later processed using the standard NLP pipeline - stopwords were removed and lemmatization was performed (using the Polish Polimorfologik dictionary [11]). As Polish is a highly inflected language, lemmatization had to be used instead of stemming. On the other hand, stopwords removal and lemmatization are not always utilized with more advanced techniques of text representation, like word or paragraph embeddings. Seminal papers that introduced those techniques do not mention stemming or lemmatization at all (cf. [10]). Therefore we have decided to test the clustering algorithm with either preprocessed corpus (i.e. with stopwords removed and lemmatization performed) or without preprocessing. Six distinct corpora for clustering were thus prepared, half of them preprocessed (those will be hereinafter denoted as C_1^P, C_2^P, C_3^P), half of them - not (hereinafter $C_1^{\neg P}$, $C_2^{\neg P}$, $C_3^{\neg P}$).
- The **processing** stage involved using the K-means clustering to group together similar documents from each corpus. The number of clusters, for the sake of the experiments, was set to 10. Visualization module uses PCA to display the clustering results.
  The following methods were used to generate document vectors as a basis of clustering:
  – TF-IDF, which used corpora C_1^P, C_2^P, C_3^P as well as $C_1^{\neg P}$, $C_2^{\neg P}$, $C_3^{\neg P}$.
  – word2vec, using the same corpora as TF-IDF. We have used the pretrained word embeddings for this part, which were generated for Polish by other research groups [12]. Those were based the National Corpus of Polish database (built using excepts from newspapers, magazines, text extracted from the internet as well as conversation transcripts) [14], in addition to Wikipedia database. Two versions of the word embedding were put under scrutiny, both holding forms for all part of speech in Polish, with vector consisting of 300 elements. Both models were trained using the negative sampling algorithm and differed in the architecture - one used CBOW, the other Skip-Gram architecture (hereinafter those will be denoted as word2vec(CBOW)

[1]The English translation of amended text comes from the Legalis legal information system, which in turn references *The Polish Law Collection* database by Translegis publishing house. The crossed-out sections were translated from Polish to English by the authors of this paper.

Table 3: Internal evaluation results of different text representation methods and corpora (bold numbers represent the best results)

Text representation	word2vec(CBOW)						word2vec(skip-gram)						doc2vec		TF-IDF					
Corpus	C_1^P	C_2^P	C_3^P	$C_1^{\neg P}$	$C_2^{\neg P}$	$C_3^{\neg P}$	C_1^P	C_2^P	C_3^P	$C_1^{\neg P}$	$C_2^{\neg P}$	$C_3^{\neg P}$	C_1^P	$C_1^{\neg P}$	C_1^P	C_2^P	C_3^P	$C_1^{\neg P}$	$C_2^{\neg P}$	$C_3^{\neg P}$
Silhouette coefficient *higher = better defined clusters*	0.16	0.14	0.1	0.17	0.23	0.09	0.16	0.29	0.13	0.18	**0.38**	0.11	0.11	0.09	0.12	0.05	0.09	0.08	0.05	0.08
Calinski-Harabaz *higher = better defined clusters*	8.03	9.75	5.72	9.74	10.77	6.28	10.27	12.01	6.04	14.17	**15.65**	6.47	3.17	2.5	2.87	2.12	2.7	2.25	1.52	2.18
Davies-Bouldin *lower = clusters better separated*	1.69	1.08	1.65	1.15	**0.54**	1.66	1.28	0.99	1.56	1.35	0.7	1.67	1.61	1.7	1.52	2.97	2.8	1.5	1.16	2.8

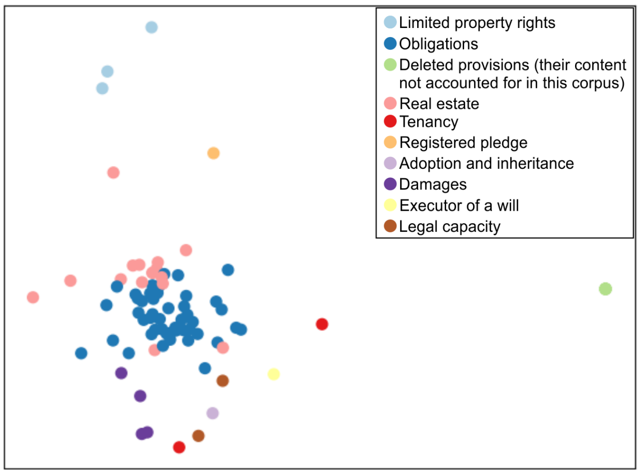

Figure 3: Clusters of amendments to the Civil Code as generated by the word2vec(CBOW) model, using the C_1^P corpus. Color-coded dots represent clusters of amending acts. Clusters types were determined by human actor.

and word2vec(skip-gram)). As word2vec holds embeddings for single words, to generate a vector that summarizes documents belonging to a corpus, the summarizing vectors were created by averaging word vectors for all the words that were present in a given amending act.

- paragraph vectors (with gensim's doc2vec implementation). Here the model was trained using commentaries to the Polish Civil Code. The texts of the commentaries were divided into 44,518 paragraphs, each consisting on average of 50 words. This corpus was used to create paragraph embeddings. The following parameters were used for embeddings generation: vector size = 1600, window = 10, training epochs = 20, training algorithm = PV-DBOW. The value of their values were determined experimentally. C_1^P and $C_1^{\neg P}$ corpora were the only ones that keep the natural flow of the text; they were the only ones tested with doc2vec embeddings.

• The results were put under scrutiny in the **evaluation** stage. There are in general two main types of verification metrics for clustering algorithms. Firstly, internal evaluation considers not a given ground truth, but the model itself. Metrics for internal evaluation presented herein include: silhouette coefficient and Calinski-Harabaz index (both evaluate how well the clusters are defined) as well as Davies-Bouldin index (assesses the separation between clusters) [13].
The external evaluation methods compare machine-generated clusters with some pre-existing evaluation gold standard, thus allowing the introduction of standard measures of precision, recall or the F-score. However, the creation of such metric in the context of this research is not a straightforward task. Obvious method of such standard creation involves classifying of existing data by a legal expert. There are however a number of concerns regarding this method. Firstly, it is necessarily subjective. Secondly, machine learning methods are conceived as means to discover latent patterns existing in the data, that are missable for humans (cf. [3]). Using human-generated gold standard therefore defeats the purpose of using machine learning methods in the first place. Thirdly, putting the subjectivity aside, creation of such gold standard is a cumbersome and tiresome task. Unfortunately, we did not have enough resources to push that venue of inquiry further. For external evaluation we have therefore settled down on qualitative methods of evaluation in place of quantitative. The clustering results, after being generated, were assessed for their distinctiveness by human actor and the best ones were selected. The grading procedure called for each result set to be reviewed and scored on 1-10 scale based on the subjective impression of results quality. The qualities such as thematic homogenity of clustered amendments, as well as their distinctiveness, were accounted for in this procedure. The relative sizes of each cluster were also considered (for example, results effecting in a single cluster holding over 75% of all amendments were considered to not be very useful).

4 RESULTS

The internal evaluation results of clustering are shown in Table 3. Generally, the word2vec (skip-gram) model achieved the best results as far as the internal evaluation results are concerned and the model worked best when it was run with the $C_2^{\neg P}$ corpus. It scored the best in terms of silhouette coefficient and Calinski-Harabaz index and well in terms of Davies-Bouldin index. Whilst preprocessing was rather detrimental to the quality of internal evaluation of results in case of various word embeddings implementations, in the case of TF-IDF metric it allowed an increase of the aforementioned quality.

In the case of external evaluation, the word2vec(CBOW) model with C_1^P corpus was ranked the highest, even though the internal evaluation results might have not pointed to that. Fig. 3 shows the visualization of the clusters as generated by this model. The results prove that contemporary word embeddings methods should be considered when preparing a clustering legal assistant. The data preprocessing phase does not have to include lemmatization, stemming or stopwords removal. However, the creation of training set and training itself remains a computationally-intensive challenge.

5 CONCLUSION AND FUTURE WORK

We have shown a proof-of-concept system capable of enhancing a lawyer with visual representation of legal change. The work presented herein was concerned with the Civil Code, however other areas of law (e.g. criminal law) should be put under scrutiny as well. Similarly, as far as the created word embeddings are concerned, we should try to create ones that use larger training sets or are more domain-oriented. Whilst this paper used traditional and well-understood methods for clustering and data dimensionality reduction, more modern techniques should be tested as well.

This work has been based on the legal change as caused by the amending acts. It should be noted that this extremely positivist (or formalistic) point of view should be supplemented with more general notions, in which the statutes themselves do not change, however the practice of officials (e.g. judges) who apply given laws does. Two examples of such practices may be given, one stemming from the practice of Polish legal system, the other based on European human rights protection system. As for the former, we have already mentioned that the Polish Constitutional Tribunal sometimes resorts to pointing out that there exists a certain interpretation of the statute that makes it compatible with the constitutional provisions. Secondly, as far as the European Convention on Human Rights is concerned, the European Court of Human Rights has on a number of occasions called it a "living instrument" and has stressed that its provisions, even if unchanged, should always be interpreted in the light of present circumstances [9]. Therefore a support system should be able to recognize the change in practice as well, which itself is a challenging problem.

This work, which is concerned with the legislative change, should therefore be viewed in the light of a broader subject of legal change. In future work we aim to employ machine learning techniques to discover and visualize changes stemming not only from the actions of the legislature, but also of other legal actors as well.

REFERENCES

[1] Timothy Arnold-Moore. 1995. Automatically processing amendments to legislation. In *Proceedings of the 5th international conference on Artificial intelligence and law.* ACM, 297–306.

[2] Timothy Arnold-Moore. 1997. Automatic generation of amendment legislation. In *Proceedings of the 6th international conference on Artificial intelligence and law.* ACM, 56–62.

[3] Kevin D Ashley. 2017. *Artificial intelligence and legal analytics: new tools for law practice in the digital age.* Cambridge University Press.

[4] Krystyna Chodorowska. 2018. *Automatic court judgment similarity analysis.* Master's thesis. Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw.

[5] Shihchieh Chou and Tai-Ping Hsing. 2010. Text mining technique for Chinese written judgment of criminal case. In *Pacific-Asia workshop on intelligence and security informatics.* Springer, 113–125.

[6] Provincial Administrative Court in Gliwice. [n. d.]. Judgment of 12th of March 2007, IV SA/Gl 1455/06.

[7] M. Kokoszczyński and G. Wierczyński. 2011. *System informacji prawnej w pracy sędziego.* Wolters Kluwer. https://books.google.pl/books?id=vmFSAwAAQBAJ

[8] Jörg Landthaler, Ingo Glaser, and Florian Matthes. 2018. Towards Explainable Semantic Text Matching. In *Legal Knowledge and Information Systems: JURIX 2018: The Thirty-first Annual Conference*, Vol. 313. IOS Press, 200.

[9] George Letsas. [n. d.]. The ECHR as a Living Instrument: Its Meaning and its Legitimacy (March 14, 2012). *Available at SSRN 2021836* ([n. d.]).

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[11] Marcin Miłkowski. [n. d.]. Polimorfologik. https://github.com/morfologik/polimorfologik

[12] IPI PAN. [n. d.]. Word2Vec. http://dsmodels.nlp.ipipan.waw.pl/

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[14] Adam Przepiórkowski, Rafal L Górski, Barbara Lewandowska-Tomaszyk, and Marek Lazinski. 2008. Towards the National Corpus of Polish.. In *LREC.*

[15] PierLuigi Spinosa, Gerardo Giardiello, Manola Cherubini, Simone Marchi, Giulia Venturi, and Simonetta Montemagni. 2009. NLP-based metadata extraction for legal text consolidation. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law.* ACM, 40–49.

[16] Franciszek Studnicki. 1978. *Wprowadzenie do informatyki prawniczej: zautomatyzowane wyszukiwanie informacji prawnej.* Państwowe Wydawnictwo Naukowe.

[17] Xin Tian, Yin Fang, Yang Weng, Yawen Luo, Huifang Cheng, and Zhu Wang. 2018. K-Means Clustering for Controversial Issues Merging in Chinese Legal Texts. In *Legal Knowledge and Information Systems - JURIX 2018: The Thirty-first Annual Conference, Groningen, The Netherlands, 12-14 December 2018.* 215–219. https://doi.org/10.3233/978-1-61499-935-5-215

[18] Ngoc Phuoc An Vo, Caroline Privault, and Fabien Guillot. 2017. Experimenting word embeddings in assisting legal review. In *Proceedings of the 16th edition of the International Conference on Articial Intelligence and Law.* ACM, 189–198.

[19] Tomasz Woś. 2016. Wyroki interpretacyjne i zakresowe w orzecznictwie Trybunału Konstytucyjnego. *Studia Iuridica Lublinensia* 25, 3 (2016), 985–995.