Document Embedding Models on Environmental Legal Documents

Samo Kralj
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
samo.kralj1@ijs.si

Živa Urbančič
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
ziva.urbancic@ijs.si

Erik Novak
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova 39, 1000 Ljubljana, Slovenia
erik.novak@ijs.si

Klemen Kenda
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova 39, 1000 Ljubljana, Slovenia
klemen.kenda@ijs.si

ABSTRACT

Finding similar documents in a big document corpus based on context has many practical applications especially in the legal sector. In this paper, our focus is on the documents related to environmental law which have been collected in a database of approximately 300k documents. We analyzed the performance of different representation models (called document embeddings) on our database and found that evaluating the results is difficult, due to the size of the database. The approaches presented can be applicable for other text datasets.

Keywords

text analysis, natural language processing, environmental law, machine learning, word embedding, document embedding

1. INTRODUCTION

When working with a large number of documents one can perform different tasks, such as finding patterns and topics within a documents, labeling documents based on their content, and finding documents that are similar to each other. These tasks can be found in multiple domains - one of them being the legal domain. There, lawyers spend hours finding documents and parts of these documents to support their legal cases.

In this paper, we present our preliminary results for finding similar documents. We employ word embeddings for creating different document representations - called document embeddings. The goal is to construct a document embedding model that enables the user to quickly find documents that are similar to a user chosen document. The documents used for evaluation are from the legal domain, but the approach can be applied to more general text datasets.

The remainder of the paper is as follows. Section 2 describes the data sources used for creating the document embeddings. Next, section 3 presents the content extraction and enrichment tool used for extracting additional document metadata. In addition, it describes different models of document embeddings using the pre-trained word2vec and fasttext word embedding models, as well as our word embedding model trained exclusively on the collected environmental documents. Section 4 presents the preliminary results of the document embedding analysis, followed by the description of future work in section 5. We conclude the paper in section 6.

2. DATA

The legal datasets used for the analysis were collected from two main sources: the first is ECOLEX [1], an online information service on environmental law led by Food and Agriculture Organization of the United Nations (FAO), the International Union for Conservation of Nature (IUCN) and United Nations Environment Programme (UNEP). The second dataset was acquired from EURLEX [3], a database of entire European Union law.

2.1 Data Acquisition

The data was collected using dedicated web crawlers. In particular, we attempted to collect as much information about each document as possible. In total, 220k and 800k different legal documents are available on ECOLEX and EURLEX datasets, respectively. The documents are ranging from the start of 20th century up until the year 2019.

There is much document metadata which is available for documents from both sources, such as the document's title, its authors, various dates (i.e. day of proposal, the day it went into force, etc.), the subject of the documents and various keywords (which are called "descriptors" in the EURLEX dataset). Bearing this in mind, there are many differences between the two acquired datasets. In this article we focus on the following: the ECOLEX dataset consists entirely of environmental law. In addition, the dataset contains much more metadata, including geospatial information (i.e. locations and countries affected by the given document), as well as a short abstract. On the other hand, the EURLEX dataset contains less metadata, but provides the complete

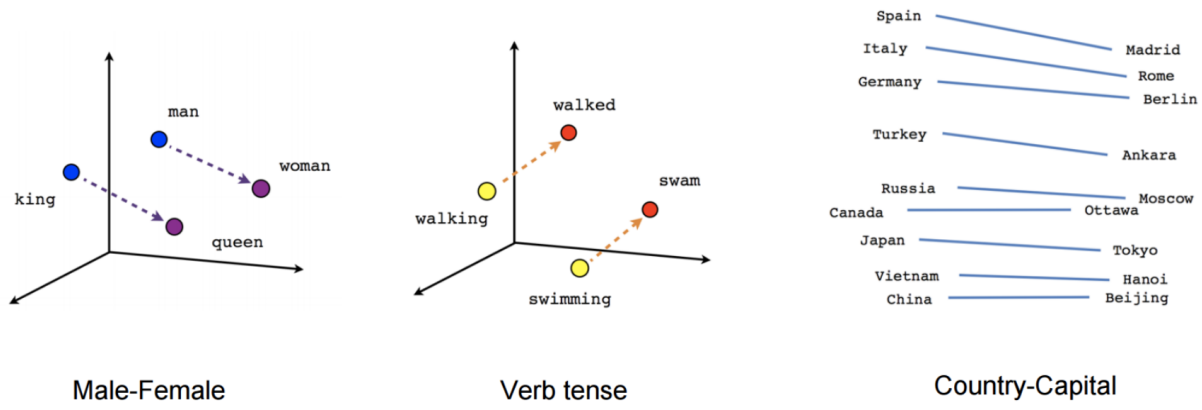

Figure 1: Word relationships captured by word embeddings. They are able to identify different relations such as male - female terms, verb tenses and other.

document content in raw text for most cases in the dataset.

Additionally, the datasets are different in two important metadata attributes: the keywords in the ECOLEX dataset and the descriptors in the EURLEX dataset. These keywords are words or phrases that best describe what the document is about. It is to be expected that documents that have similar keywords are also similar in content. While keywords and descriptors serve similar purpose in the respective dataset, they are not the same. A particular keyword might not be included in the descriptors word corpus and vice versa. Furthermore, keywords that describe some document are different from descriptors that describe a similar document.

2.2 Dataset Statistics

Out of the 800k EURLEX documents collected, 300k were filtered out based on whether the full text of the document is available in English, German and Slovene language. Since we are interested in documents dealing with environment, further filtering was done using document descriptors, keeping only documents with at least one environmental descriptor. In total, approximately 75k documents were considered to be appropriate for our analysis.

Since the ECOLEX documents are already focused only on environment, no further filtering was required.

3. METHODOLOGY

In this section we describe our approach for analyzing a big corpus of documents, namely document embeddings. Even though a lot of pre–processing was necessary to prepare the documents' texts for later use (making sure all letters are lowercase, stripping the punctuation from the text, removing words that appear frequently in the language – for example prepositions), we will not discuss this further in the paper. We also appended additional information to the documents using the content extraction and enrichment tool, which we describe in section 3.1. Further, we focus on word embeddings and different methods of how to use them to create document embeddings in sections 3.2 and 3.3, respectively.

3.1 Content Extraction and Enrichment Tool

To enrich the documents, we annotated all documents using the InforMEA ontology, a hierarchy of environmental terms. Also, document's text was sent into Wikifier - a web service that extracts major Wikipedia concepts from the text. The resulting concepts were added to the document's metadata. These annotations add additional keywords and concepts to the document, improving our representation of documents that may have poor keywords representation, and adds additional metadata to documents that already had a good keyword representation but might be missing some important keyword.

3.2 Word Embedding

In natural language processing, word embedding has been a popular method for representing textual data in the past years. It is a model trained on character n–grams of the word and on what is called context: the target word's neighboring words. In the model, the words are represented as vectors – usually in high-dimensional space - where the inherited geometric relations mimic relationships between words in the language. Word embeddings are able to capture both syntactic and semantic information about the word. Some of the relationships between words captured by word embeddings are shown in figure 1.

The most popular word embedding models available to the public are word2vec [10] and fasttext [9]. What sets them apart is what they consider to be an atomic embedding element: word2vec considers a word to be the smallest part of language to embed, while fasttext uses character n-grams as well - it embeds them as if they were words. Because of this we can extract embeddings for out-of-vocabulary terms, providing embeddings of rare and previously unseen words. We decided to employ two models: a) the pre-trained fasttext model for the English language, and b) the model trained on our database of environmental legal documents. In addition, aligned vectors for 44 languages [6, 4] are available, which will be used in the future work to enable cross-lingual search of documents.

3.2.1 Training a Word Embedding Model

One of the word embedding models we employed was trained on our database. Instead of having a a large vocabulary of pre-computed word embeddings trained on Wikipedia and Common Crawl, this newly trained model is trained on documents from a more specific domain - resulting in a vocabulary limited to the topics found in the documents within the corpus (e.g. in our case environmental law). This approach might improve the performance in cases when the language is domain specific.

The new fasttext model has been trained using the gensim library. In order to be consistent with the pre-trained fasttext model, we decided the trained model should provide word embeddings as 300-dimensional vectors. We set a threshold of 4 appearances to avoid noise. In comparison with the vocabulary of the pre-trained fasttext model, the vocabulary of our model is 5 times smaller, consisting of approximately 500k tokens. Its initial performance is described in section 4.1.

3.3 Document Embedding

To be able to retrieve and compare documents, they must first be represented in a form that the machine will be able to understand. Similar as for words, the most common form of document representation is as a vector. We chose to represent a single document as an average of word embeddings of words found in that document. In other words, let $W = \{w_1, w_2, \ldots, w_n\}$ be a list of words that appear in a document, and let $\{x_1, x_2, \ldots, x_n\}$ be the list of word embeddings associated with the words in the document. Then the document embedding is calculated by following the equation

$$d = \frac{1}{|W|} \sum_{w_i \in W} x_i.$$

Further, we considered some other embedding methods. The first approach is to define the document embedding as an average of word embeddings of only the most significant words, namely document descriptors for the EURLEX dataset or keywords for the ECOLEX dataset. The reasoning behind it is that it might speed up the calculation, but it comes at a cost of neglecting a lot of information we have about documents and the possibility of reducing the quality of the result. To avoid the listed downsides, we propose a combined embedding, which would be defined as a linear combination of two embedding methods described above. This embedding unfortunately loses the advantage of fast computation, but it does give more weight to more important words of the document. In order to decide which method performes better, we performed some analysis which is described in section 4.

Once the document embeddings are calculated - depending on the chosen method and word embedding model - we are able to find semantically similar documents by calculating the distance of their embeddings. Figure 2 shows the mapping of the document embedding into the 2-dimensional space using the t-SNE algorithm [8].

4. PRELIMINARY RESULTS

We split our analysis in two parts. In section 4.1 we tested various document embedding models based on the choice of word embedding models. In addition, we perform an analysis using different approaches of constructing document embeddings given a pre-trained fasttext word embedding model, which is described in 4.2.

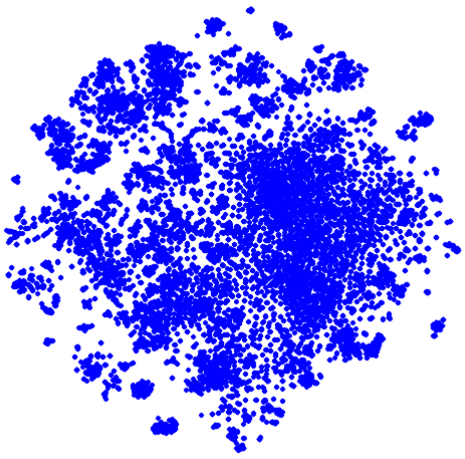

Figure 2: Planar projection of document embeddings of the first 15k English legal documents in the EURLEX corpus. Our assumption is that similar documents have similar embeddings and therefore form clusters, which we evaluated manually.

4.1 Performance of Different Word Embedding Models

When deciding which document embedding model to use, the choice of word embedding model is very important. We are interested in which of the two word embedding models described in section 3.2 produces a better document embedding model. In this part of the analysis we chose to construct document embeddings as the average of word embeddings of words appearing in the text of the document.

Manually checking the results for some arbitrary examples we noticed that the newly trained word embedding model outperforms the pre-trained one when the source document is not particularly similar to any other document in the database. Our observations are based on using only the model trained with parameters described in section 3.2.1. Further analysis of training parameters will be performed in the future.

4.2 Performance of Different Document Embedding Models

It is hard to evaluate and compare different document embedding models. We performed manual checking and found satisfactory results in some cases. To test the model we picked a random document and found the k "most similar" documents using the k-nearest neighbors algorithm [5] and the cosine distance.

What follows is an example of such a search for $k = 5$ using a document embedding model based on the text of the document (the title is not included). The first item is the title

of the source document, while the rest are the titles of the most similar documents:

1. **Source:** European Convention for the protection of animals kept for farming purposes.
2. Convention on the protection of the Mediterranean Sea against pollution (Barcelona Convention).
3. Protocol concerning Mediterranean specially protected areas.
4. Protocol for the protection of the Mediterranean Sea against pollution from land-based sources.
5. Protocol of amendment to the European Convention for the protection of animals kept for Farming purposes.

The given results are quite good, but it seems like the document on the fifth position is the most similar to our source document - showing that the presented model still has potential for improvement. Figure 3 shows the result of the search for 10 most similar documents using the text of the document in document embeddings. Marked with the red dots are the documents acquired from the search results.

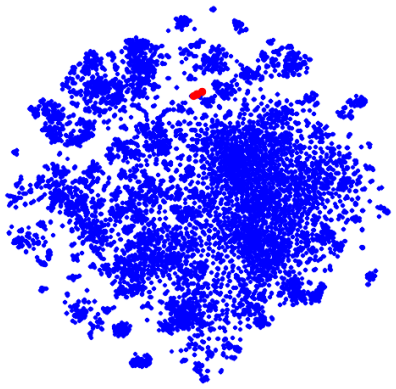

Figure 3: Projection of a document embedding model using the words from documents text as a representation. Red dots represent the 10 document embeddings that are closest to the embedding of the source document.

5. FUTURE WORK

Manually checking the complete corpus of a few 100k documents is time consuming. The amount of documents is huge and we also do not have the ability to tell how good the results are. There is no easy way to define a metric that could compare how well different models perform. Therefore, we will try to evaluate and improve our model using the users feedback. We will develop a service which will enable the user to perform queries for the legal documents. Each time a user makes a query, the system will note the documents that the user checked. With this feedback we will be able to update and improve our model.

In addition, we will consider another distance metric called the Word Movers Distance [7] when calculating the document similarity using word embeddings.

6. CONCLUSION

Word embeddings and document embeddings have proven to be useful when performing analysis on a large textual dataset. The available word embedding models on which we based our research - word2vec and fasttext - are exhaustive and easy to use. What we have done so far has given satisfactory results on recognizing similar documents, which we hope to improve with further work, especially by finding a model that will fit our dataset of environmental legal documents best and then developing it based on user feedback.

7. ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the European Unions Horizon 2020 project enviroLENS under grant agreement No 821918 [2].

8. REFERENCES

[1] Ecolex - a gateway to environmental law. `https://www.ecolex.org/result/?q=&xdate_min=&xdate_max=`. Accessed: 2018-12-20.

[2] EnviroLens project. Accessed in: August 2019.

[3] Eur-lex - access to european law. `https://eur-lex.europa.eu/homepage.html`. Accessed: 2019-02-25.

[4] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics 5* (2017), 135–146.

[5] Cover, T., and Hart, P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory 13*, 1 (January 1967), 21–27.

[6] Joulin, A., Bojanowski, P., Mikolov, T., Jégou, H., and Grave, E. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018).

[7] Kusner, M. J., Sun, Y., Kolkin, N. I., and Weinberger, K. Q. From word embeddings to document distances. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37* (2015), ICML'15, JMLR.org, pp. 957–966.

[8] Maaten, L. v. d., and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research 9*, Nov (2008), 2579–2605.

[9] Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)* (2018).

[10] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2* (USA, 2013), NIPS'13, Curran Associates Inc., pp. 3111–3119.