Efficient Neural-based patent document segmentation with Term Order Probabilities

Danilo S. Carvalho * and Minh-Le Nguyen

School of Information Science - Japan Advanced Institute of Science and Technology
Nomi City, Ishikawa - Japan

Abstract. The internationally growing trend of patent applications puts great pressure on the agents involved in managing this kind of information and creates a demand for efficient and effective patent analysis methods. This work presents a computationally efficient approach for patent document segmentation based on structured ANNs and a simple distributional semantics composition method. The conducted experiments indicate effectiveness of the approach, which benefits a wide array of patent processing techniques that work upon structured inputs.

1 Introduction

The fast ascension of *Intellectual Property* as a critical element in domestic and international trade, as well as in academics, puts increasing pressure in all agents involved in creation and management of this kind of asset. Such agents include businesses, universities, patent and trademark offices, and courts of law. In the case of patents, managing information related to innovation is a very difficult task, comprising a thorough analysis of vast amounts of academic and legal documents in search for potential infringement or for new innovation avenues.

One of the fundamental steps in analyzing patent documents is determining its basic structure: a typical patent is divided into sections, e.g, *abstract*, *description*, *claims*. The information contained in each section can be used for different purposes, such as the use of abstracts for document summarization and classification. Separating those sections is a task known as *patent document segmentation*, which is the focus of this work. Currently, the biggest patent offices (USPTO, EPO, among others) do all document processing electronically, meaning the documents are segmented from the start. However, there are several patent offices in which paper forms are still in use. Those, when digitized through OCR[1], result in unstructured documents, for which the method described in this paper can be applied.

This work presents an efficient approach to patent document segmentation, through the use of a computationally inexpensive method of combining distributional semantic representations (word embeddings) into sentence representations. Such representations are then used as features for a *Structured Perceptron* sequence classifier for sentence tagging, from which section boundaries can be determined. Experiments conducted with a subset of the public USPTO document database indicate that the composition method keeps word order information,

[*] This work was supported by CNPq Brazil scholarship grant.

[1] Optical Character Recognition

thus improving the segmentation performance when compared to a bag-of-words approach using only the word embeddings.

This remainder of paper is organized as follows: Section 2 presents previous research related to the topic of patent document segmentation. Section 3 describes some fundamental concepts involved in this work. Section 4 explains the segmentation approach in detail. Section 5 describes the experimental evaluation and discusses the evaluation results. Finally, Section 6 offers a summary of the findings and some concluding remarks.

2 Related work

Recent research on patent document segmentation for Information Extraction can be found in the works of Sheremetyeva [1], who presented a two level procedure for decomposition of section and claim structures grounded on deep linguistic analysis, and also Brugmann et. al. [2], who presented a complete document analysis system for patents, featuring several different techniques, ranging from document segmentation (section identification, claim spotting) to claim description analysis, entity recognition and document summarization. For comparison purposes, the segmentation method applied in this work covers both the first level and third level's first stage of the segmentation hierarchy proposed in [2], but using a Structured Neural Network instead of CRFs combined with a set of heuristics. A direct attempt to structurally compare patent documents is presented by Huang et. al. [3], using Structured Self-Organizing Maps (SOM).

3 Fundamental concepts

3.1 Patent Information Processing

The general goal of information processing on patent documents is to aid the work of patent examination professionals, and also to reduce the chance of patent rejection or litigation by applicants and grantees. Document analysis involves understanding of the concepts described in the patent and how they relate to compose a *patent scope*, which defines what will be protected by law.

A patent document is composed of sections, each one having a different role in describing the invention being protected, some of them being optional. Typically: header, abstract, description, claims and illustrations. A patent information processing system must break the document into its sections, examine the contents, and expose its characteristics, in a way that facilitates automated scope comparison or the retrieval of documents by their scope. These tasks are made difficult by the fact that document conventions change over time and syntactic patterns also differ from those used in common language, making the use of standard parsers less reliable and motivating the use of semantic information.

3.2 Distributional semantics, word and sentence embeddings

The concept of *distributional semantics* is based on the notion that words are always used in a context, and the context defines their meaning. Therefore, the meaning of a word can be defined as a function of its neighbors (co-occurrence). This definition allows representations of words in a chosen vector space, and such representations are called *embeddings*. They enable the use of vector operations on words, such as comparison by cosine similarity, and also solve the data sparsity problem of large vocabularies, working as a dimensionality reduction method. The distributional approach also presents an attractive option for compact sentence representation, often through the composition of word embeddings [4] [5]. The most popular distributional representation approaches for sentences offer good performance on semantic relatedness and similarity tasks [4], but have a considerable computational cost compared to their word counterparts, which poses a problem for their application to big corpora, such as patent databases.

4 A simple architecture for patent document segmentation

As a starting point, the segmentation task was defined as a membership problem, in which the elements are the document sentences. A sentence may only be part of a single section and the sections are sequences of sentences, so that one or two sentences in each one are boundaries. Thus, a tagging scheme including both membership and boundary information was used. Fig. 1 illustrates the tagging.

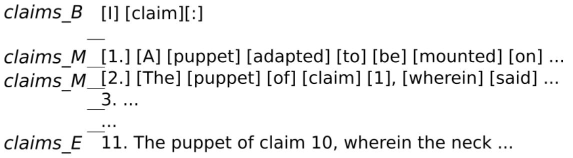
claims_B [I] [claim][:]
claims_M [1.] [A] [puppet] [adapted] [to] [be] [mounted] [on] ...
claims_M [2.] [The] [puppet] [of] [claim] [1], [wherein] [said] ...
3. ...
...
claims_E 11. The puppet of claim 10, wherein the neck ...

Fig. 1: Example of sentence tagging for the claims section of a patent document. claims_(B/M/E) are the beginning, middle and end of the section respectively.

Assuming a reliable semantic representation of a sentence's content, the next step was to choose a sequence classifier that could take such representations as inputs and predict the correct tags, specially the boundary ones. The use of distributional embeddings makes the inputs well suited for a Neural Network based approach. Considering the low computational cost preference, the Structured Perceptron [6] was chosen over more complex models, such as Recurrent Neural Networks (RNN). The Structured Perceptron is an extension of the traditional Perceptron algorithm in which the feature function $\Phi(x, y)$ takes as input both the original input x and a candidate prediction y. The prediction is calculated with $\hat{y} = argmax(w^\top \Phi(x, y))$ and the weights $w^\top$ are updated using the incorrect answer with the highest score y': $w \leftarrow w + \Phi(x, y') - \Phi(x, \hat{y})$.

4.1 Inexpensive sentence embeddings: Term Order Probabilities

Considering the structural properties of syntactic constrained sentences typically found in patent documents, a method was developed to quickly obtain representative vectors of entire sentences, aimed at speeding up the segmentation process. It rests on the assumption that the higher structural regularity would decrease the amount of information lost due to the use of a less accurate method.

The method consists in calculating the probability $P(t_1, t_2)$ of any pair of terms (words, n-grams) t_1 and t_2 appearing in this specific order in the sentence. This value was called *Term Order Probability* (TOP) and can be easily calculated using the formula $P(t_1, t_2, d) = \frac{\#(t_1,t_2,d)}{\#(t_1,t_2,d)+\#(t_2,t_1,d)}$, where $\#(X)$ is the number of occurrences of X in the reference corpus and d is the maximum distance between t_1 and t_2. After calculating TOP for a corpus with n terms, the result is a matrix $\mathbf{P}^{n \times n}$ which is very sparse and can be efficiently stored and accessed. This information is useful for including basic structural information from a sentence in simple vector operations, as in the sentence embeddings described next.

To generate sentence embeddings using TOP, the following formula is used:

$$\mathbf{s} = \frac{\sum_{i=0}^{k} t_i + \sum_{i,j=0:i<j}^{k} (t_i + t_j) * (1 - P(t_i, t_j))}{k + \sum_{i=0}^{k} k - i} \tag{1}$$

where t_i are the term embeddings and k is the length of the sentence. The resulting vector is the sum of the TOP-weighted combinations of all embedding pairs in the sentence. The range of j may be limited to create a fixed size window of distance d for each term, improving efficiency in longer sentences. The idea behind this formulation is that the contribution of each term to the sentence embedding is weighted by an "attention index" $(1 - P(t_i, t_j))$, representing how unlikely the term is to appear in that context. In this way, uncommon patterns have a higher contribution, helping to distinguish even between similar sentences. While not as precise as Machine Learning-based sentence embedding methods, such as [4], the cost of using TOP is much lower. The TOP matrix is calculated only a single time and can be built incrementally. The sentence embeddings obtained in this way are then used as inputs for the Structured Perceptron. Figure 2 illustrates the processing flow of a sentence in the document.

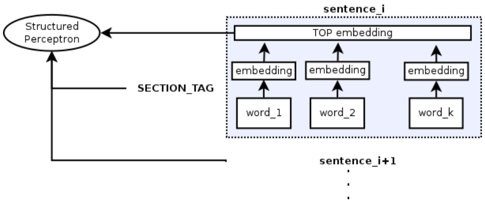

Fig. 2: Sentence processing flow. Each TOP composed embedding and sentence label serve as input for the Structured Perceptron ANN.

5 Experimental results

5.1 Data and experimental setup

Experimental data was obtained from the United States Patent and Trademark Office (USPTO) patent repository, made available by Google [7]. USPTO documents were chosen due to their availability and the fact that all recent documents have annotations up to the level of claims, providing us with ground truth data for the tests. A set of 2000 patent documents from January to February 2015 (1K documents for each month) was used in the tests, totaling about 80 million sentences and 132 thousand terms, after n-gram modeling (N $\leq$ 3).

The documents were split into sentences and each sentence was labeled according to the section and position it occupied, e.g.: CLAIM_(B/M/E) for the beginning, middle and end of the claims section, respectively. Optional sections were not included. Training and prediction task was performed using the *Structured Perceptron* algorithm [6], taking the sentence embeddings of each document as input, with parameters: *decode* = *viterbi*, *lr* = 0.1, *iter* = 10 (implementation defaults). For the term embeddings, *word2vec* [8] was trained over the USPTO corpus for the set of documents from January to March 2015, with parameters: $d = 200$, $cbow = 1$, $window = 10$, $neg = 25$, $iter = 15$, $hs = 0$ and $sample = 1e-5$. Sentence embeddings were generated by using the formula described in Section 4.1, and also by sum and average of the *word2vec* vectors, and by using the *doc2vec* implementation of the *Paragraph Vectors* [4] algorithm. *Doc2vec* parameters were set the same as *word2vec*, except $dm = 0$ and $iter = 5$ to make training time practical, so all embeddings have dimensionality = 200. *TOP* matrix calculation and doc2vec training were done over the same corpus section as word2vec. Accuracy was used as performance measure, calculated by taking the average ratio of correct predictions per class (tag) in the document collection. A document-wise 10-fold cross validation was performed 3 times, and the average results recorded. The term window size j for sentence embedding was adjusted between 2 and 8, running the cross validation once per value until finding the best. The sum and average approach was used as a baseline. A running time measurement was also done, separated in training and test times. The test times represent the average CV fold time. TOP training time includes *word2vec* training time. The experiments were run on a Xeon 2GHz CPU (6 cores), 64GB of RAM computer. *Word2vec* and *doc2vec* training were done with 4 threads, all the rest being single-threaded.

5.2 Results

Experimental results are shown in Table 1.

Table 1: Results from the document segmentation test. *sum&avg* means sum and averaging of all vectors in a sentence. *ws (window size)* is the maximum term lookahead applied to the sentence embedding formula in *TOP*.

Method	Period	Accuracy	Train time (min)	Test time (min)
Word2Vec sum&avg	Jan 2015	92.8%	116	0.2
Doc2Vec [4]	Jan 2015	93.4%	535	1.0
W2V TOP (ws = 4)	Jan 2015	**98.5%**	187	0.4
Word2Vec sum&avg	Feb 2015	92.7%	116	0.2
Doc2Vec [4]	Feb 2015	89.0%	535	1.8
W2V TOP (ws = 4)	Feb 2015	**97.4%**	187	0.6

The results indicate that the structural information included by *TOP* improved the segmentation performance, while keeping a low computational cost. The obtained accuracy is adequate for real-world applications and is compatible with the findings of Brugmann et. al. [2], which reported a F1-score of 0.93 in this task for European patents, and Sheremetyeva [1] which reported a 100% accuracy result through supertag-based parsing, albeit with a much smaller document set (25 documents), however, a direct comparison could not be made for [2] and [1] due to the use of different sets of documents and the authors being unable to find a public implementation of both methods. The reduced performance of *doc2vec* can be explained by lack of training data, which is a dominant factor for that method. It is expected to increase with a larger training set, with training time increasing as well, posing another advantage to *TOP*.

Analysis of the error cases showed that most prediction mistakes occur in the transition from the last claim in the claim section to the beginning sentence of following section, which is always an optional one. A typical example is an illustration section beginning without a title and with a sentence structure similar to a claim, since claims may also refer to illustrations. A second most common error case is found in untitled transitions abstract → description. Training with the optional sections would be an appropriate way to deal with the former cases, while the latter indicates a need for either a better representation or classifier.

6 Conclusion

Improvements on document structuring techniques have a discrete but wide effect on the patent analysis landscape, for which there is a growing demand of efficient and effective processing methods.

This work presented a simple, yet effective architecture for patent document segmentation, based on Structured Neural Networks and a word embedding composition method for sentence representation. The composition method uses pre-computed values from the corpus, called Term Order Probabilities (TOP). This method was developed to facilitate the extraction of structural information from sentences and the generation of sentence embeddings in larger corpora, which are difficult to process using more complex sentence embedding methods.

The experimental results indicate a positive impact on the document segmentation task and point us to immediate future work, which includes expanding our patent database with other sources, e.g. European and Japanese patent office documents. Evaluating parameter sensitivity issues in the current classifier and testing with other structured classification methods, e.g. *LSTM*, is also planned.

Acknowledgment

This work was supported by JSPS KAKENHI Grant JP15K16048 and JAIST research grant.

References

[1] Svetlana Sheremetyeva. Automatic text simplification for handling intellectual property (the case of multiple patent claims). *COLING 2014*, page 41, 2014.

[2] Sren Brgmann et al. Towards content-oriented patent document processing: Intelligent patent analysis and summarization. *World Patent Information*, 40:30 – 42, 2015.

[3] Su-Hsien Huang, Hao-Ren Ke, and Wei-Pang Yang. Structure clustering for chinese patent documents. *Expert Systems with Applications*, 34(4):2290 – 2297, 2008.

[4] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196, 2014.

[5] Hang Yin, Chunhong Zhang, Yunkai Zhu, and Yang Ji. Representing sentence with unfolding recursive autoencoders and dynamic average pooling. In *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*, pages 413–419. IEEE, 2014.

[6] Michael Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. ACL, 2002.

[7] USPTO. Google uspto patents. (https://www.google.com/googlebooks/uspto-patents.html), 2016. [Online; accessed 15-July-2016].

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.