Enriching Word Embeddings for Patent Retrieval with Global Context

Sebastian Hofstätter[1], Navid Rekabsaz[2], Mihai Lupu[3], Carsten Eickhoff[4], and Allan Hanbury[1]

[1] Vienna University of Technology, Vienna, Austria
{sebastian.hofstaetter,allan.hanbury}@tuwien.ac.at
[2] Idiap Research Institute, Martigny, Switzerland navid.rekabsaz@idiap.ch
[3] Research Studios Austria, Vienna, Austria mihai.lupu@researchstudio.at
[4] Brown University, Providence, USA carsten@brown.edu

Abstract. The training and use of word embeddings for information retrieval has recently gained considerable attention, showing competitive performance across various domains. In this study, we explore the use of word embeddings for patent retrieval, a challenging domain, especially for methods based on distributional semantics. We hypothesize that the previously reported limited effectiveness of semantic approaches, and in particular word embeddings (word2vec Skip-gram) in this domain, is due to inherent constraints on the (short) window context that is too narrow for the model to capture the full complexity of the patent domain. To address this limitation, we jointly draw from local and global contexts for embedding learning. We do this in two ways: (1) adapting the Skip-gram model's vectors using global retrofitting (2) filtering word similarities using global context. We measure patent retrieval performance using BM25 and LM Extended Translation models and observe significant improvements over three baselines.

1 Introduction

Distributed representations of semantic and syntactic term content are surging in popularity. Several recent studies [4,7,17,18,19,20,21,22] focus on novel approaches to representing words in a vector space and show promising retrieval results in domains such as Web, news, and health search.

Prior art search (or *patent retrieval*) is a challenging retrieval domain. The nature of patent text has been shown to be a source of difficulty for retrieval models that perform very well on other domains [9]. In fact, the effectiveness of semantic resources, especially distributional semantics for patent retrieval has been disputed altogether [8].

In this paper, we revisit this problem in light of recent advances in word embedding learning for document retrieval. We hypothesize that the limited effectiveness of state-of-the-art word embeddings (e.g. word2vec Skip-gram [10]) is due to their focus on local word context and that this is too narrow to capture the complexity of the patent domain language. Since fully extending contexts to the document level has also been shown not to perform well [8], we will investigate the combination of both local and global (document) contexts for embedding learning. We show that by drawing from these complementary sources of information, we can significantly improve performance in terms of recall-based measures that are central in this domain.

We use the Extended Translation variants [18] of $BM25$ and language models (LM) [14], referred to as $\widehat{BM25}$ and $\widehat{LM}$ to factor statistical semantics into the retrieval models. We examine the retrieval effectiveness using a word2vec Skip-gram embedding (based on a local window context) and observe that using $\widehat{BM25}$ and $\widehat{LM}$ with similar words from Skip-gram leads to a mild, yet statistically insignificant improvement in retrieval performance in the patent domain. The use of global context was previously suggested as an additional filter method [17] in other domains. We extend this hypothesis to the patent domain and additionally create a new vector representation based on local and global context. We employ the document-wide context of words using Latent Semantic Indexing (LSI).

To combine LSI and Skip-gram based word similarities we study two methods: (1) retrospectively adapting the Skip-gram model's vector representations based on the LSI-induced word similarities using Retrofitting [5]. (2) Inspired by the Post-Filtering method [17], we filter the Skip-gram model's result according to the LSI model similarities. In addition, motivated by previous studies [5,12], we examine the effects of using explicitly curated semantic lexicons (e.g., WordNet). To this end, we propose two methods to combine LSI-induced similarity information and semantic lexicons.

We evaluate the methods on the CLEF-IP 2013 benchmark [13] and show a significant improvement in comparison with $BM25$ and LM as well as $\widehat{BM25}$ and $\widehat{LM}$ using Skip-gram and LSI separately.

This study fits into the larger category of research using or learning semantic resources for retrieval: some use pseudo-relevance information for training per-query word embeddings [4] or generic query embeddings [21]. Other studies follow a supervised approach to learning IR-specific word representations [19,20] from relevance judgments. In contrast to these studies, our retrofitting approach learns a generic word embedding (no per-query overhead) and does not require industry-scale amounts of relevance judgments or sample queries.

2 Background

2.1 Retrofitting

Retrofitting [5] is an efficient post-processing method to adapt vector representations of existing word embeddings based on word-word similarities provided by a secondary resource. The method modifies the original vector representations by optimizing the following objective function:

$$\Psi(V) = \sum_{t \in T} \left[\alpha_t \left\| v_t - \widehat{v}_t \right\|^2 + \sum_{t' \in R(t)} \beta_{tt'} \left\| v_t - v_{t'} \right\|^2 \right] \tag{1}$$

where $\widehat{v}$ is the original vector and $v \in V$ denote its retrofitted vectors, T is the set of words in the embedding, and $R(t)$ is the set of similar words in the external resource. α_t represents the weight of the original vector of word t, and $\beta_{tt'}$ represents the similarity weight between the words t and t' in the external resource.

In order to minimize $\Psi(V)$, the derivative of Eq. 1 is set to zero, resulting in the following vector update formula:

$$v_t = \frac{\sum_{t' \in R(t)} \beta_{tt'} v_{t'} + \alpha_t \widehat{v}_t}{\sum_{t' \in R(t)} \beta_{tt'} + \alpha_t} \tag{2}$$

As shown in the formula, with each update v_t comes closer to the related vectors $v_{t'}$, where relatedness is defined and measured by the external resource. The retrofitting method iteratively updates the vectors with Eq. 2 until convergence.

2.2 Extended Translation Models

Rekabsaz et al. [18] introduce Extended Translation models for several probabilistic retrieval models (among which BM25 and LM) as a variant to the translation LM [3], providing a robust way of using word embeddings for document retrieval. The authors consider a form of term-term relation, based on the underlying concepts of each term, where the concepts are extracted from an embedding model. The Extended Translation models therefore, instead of counting the occurrences of a term, count the occurrences of the term's concepts in the documents. Based on this idea, they define the extended tf of a query term t in a document d as:

$$\widehat{tf_{t,d}} = tf_{t,d} + \sum_{t' \in R(t)} P_T(t|t') tf_d(t') \tag{3}$$

where $P_T(t|t')$ is the translation probability, and $R(t)$ is the set of similar terms, both captured from a word embedding. In addition to $\widehat{tf}$, the Extended Translation models use updated versions of other components (i.e. document length, collection and document frequency), calculated in accordance to the changes in term frequency.

3 Methodology

The focus of this paper lies on the source and measurement of global context used in the retrieval, rather than the retrieval models themselves. In this section, we propose different models to gauge the necessary word-word similarities.

SkipGram, LSI These two baseline methods use a set of related words obtained from a word2vec Skip-gram embedding, and an LSI embedding, respectively. For each model we empirically determine a threshold on the similarity values between words by evaluating a parameter sweep over the threshold parameter.

Retro(⋆) This method applies retrofitting on a Skip-gram word embedding. The input resource ⋆ can be any external resource defining a similarity relation between words. Similar to [5], we set $\alpha_t = 1$ in Equation 1, and normalize the values of β so that the sum of $\beta_{tt'}$ for word t is equal to one:

$$\beta_{tt'} = \frac{s_{tt'}}{\sum_{t'' \in R(t)} s_{tt''}} \tag{4}$$

where $s_{tt'}$ is the similarity score between the words t and t', given by the external resource. If the input resource is also a word embedding scheme (i.e., LSI), a second threshold (for selecting LSI similarities) is required in order to define $R(t)$.

PostFilter(⋆) This method filters the set of related words of SkipGram ($R(t)$), removing any words that do not also appear in the set of related words of the external resource $R^*(t)$. Hence, PostFilter is defined by the conjunction of both sets: $R(t) \cap R^*(t)$. In general, PostFilter models follow a conservative approach by considering two words related only when both the SkipGram and the external model agree.

`ExtRetro(*,*)` The Extended Retrofitting model exploits two input resources for the retrofitting procedure. The model extends Eq 2 as shown in the following:

$$v_t = \frac{\gamma \sum_{t' \in R^1(t)} \beta^1_{tt'} v_{t'} + (1-\gamma) \sum_{t' \in R^2(t)} \beta^2_{tt'} v_{t'} + \alpha_i \widehat{v}_t}{\sum_{t' \in R^1(t)} \beta^1_{tt'} + \sum_{t' \in R^2(t)} \beta^2_{tt'} + \alpha_i} \tag{5}$$

where the superscripts on $R(t)$ and β indicate the corresponding similarity models, given as input. In our experiments we set γ to 0.5 to enable both resources to have an equally strong influence.

`PFRetro(*,*)` The Post-Filter Retrofitting model combines the information of two external resources for the final set of related terms. It applies the `PostFilter` using the first input on the results of the `Retro` model, retrofitted by the second input resource.

4 Evaluation and Results

This section describes our experiment setup, presents and discusses the evaluation results, and finally analyzes the robustness of the methods.

4.1 Experiment Setup

Benchmark and Indexing We conduct experiments on the CLEF-IP 2013 Claims to Passage task [13]. The collection contains approximately 2.6 million patent documents, and 50 query topics. Similar to Anderson et al. [1], we formulate the queries by selecting the top 100 words in the query documents with highest $tf\,idf$ weights. We conduct the evaluation on the document level using the standard evaluation metrics of the task, namely MAP, PRES@1000, and RECALL@1000. For the retrieval we use Lucene and our implementation of $\widehat{BM25}$ and $\widehat{LM}$[5]. As suggested by previous studies [1,8], we do not apply stemming.

Similarity resources We create the Skip-gram word embedding with 300 dimensions on the complete CLEF-IP corpus using Gensim [15]. We use a window of 5 words, negative sampling of 10, down sampling of 10^{-5}, 20 epochs, and filtering words with frequency less than 100. We experiment with two types of external resources for word similarities: Document-context Latent Semantic Indexing (LSI), and semantic lexicons. The LSI word embedding is created on the CLEF-IP text corpus, following the approach in Rekabsaz et al. [17]. Similar to Faruqui et al. [5], we use four semantic lexicons: FrameNet [2], PPDB [6], only synonyms of WordNet [11] (WN.synonyms), and WordNet with synonyms, hypernyms and hyponyms (WN.synonyms+).

Baselines We use three baselines to compare the retrieval performance of $\widehat{BM25}$ and $\widehat{LM}$, when using the word similarity methods: The first are the standard $BM25$ and LM (without adding any semantic information), which we refer to as `None`, the second are $\widehat{BM25}$ and $\widehat{LM}$ using the local-context `SkipGram` method, studied in previous work [18,16], and the third are also $\widehat{BM25}$ and $\widehat{LM}$ but using the global-context `LSI` method. We measure statistical significance of differences of the results using a two-sided paired t-test with $p < 0.05$.

[5] Our code and the Lucene extensions are available at: *github.com/sebastian-hofstaetter/ir-generalized-translation-models*

Table 1: Evaluation results of various word similarity methods on the CLEF-IP 2013 collection. Statistical significance to baselines: †: `None`, ρ: `SkipGram`, ω: `LSI`

Word similarity method	$\widehat{BM25}$			$\widehat{LM}$		
	MAP	**PRES**	**RECALL**	**MAP**	**PRES**	**RECALL**
`None`	0.184	0.607	0.703	0.200	0.669	0.755
`SkipGram`	0.207†	0.615†	0.679	0.200	0.665	0.758
`LSI`	0.191	0.650†ρ	0.737ρ	0.205	0.676	0.752
`Retro(FrameNet)`	0.206	0.633†	0.698	0.188	0.661	0.762
`Retro(WN.synonyms)`	0.206	0.610	0.705	0.208	0.651	0.717
`Retro(WN.synonyms+)`	0.180	0.597	0.674	0.207	0.638	0.754
`Retro(PPDB)`	0.194	0.625	0.715	0.240†$\rho\omega$	0.667	0.758
`PostFilter(LSI)`	**0.247**†$\rho\omega$	0.638†ρ	0.733ρ	0.228†$\rho\omega$	0.689	0.785
`Retro(LSI)`	0.238†ω	0.639†	0.733ρ	0.221	**0.698**	**0.812**†$\rho\omega$
`ExtRetro(LSI, PPDB)`	0.239†$\rho\omega$	0.624	0.733ρ	0.227†$\rho\omega$	0.669	0.765
`PFRetro(LSI, PPDB)`	0.246†$\rho\omega$	0.643†ρ	0.733ρ	0.218†ρ	0.686	0.788$\rho\omega$

Parameter Settings The Dirichlet prior μ of the LM and also b, k_1, and k_3 for $BM25$ are shared between all method variants, hence we use the same set of values suggested by Rekabsaz et al. [18]. We explore cosine similarity threshold values to select similar words in the range of $[0.6, 1]$ with steps of 0.01. We explore LSI threshold values in the range of $[0.5, 0.9]$ with steps of 0.02. The final results are reported by applying 5-fold cross validation.

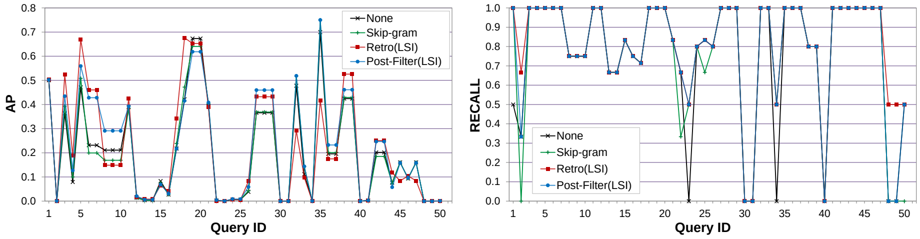
4.2 Results and Discussion

Table 1 reports retrieval performance of $\widehat{BM25}$ and $\widehat{LM}$, comparing the methods presented in Section 3. Contrasting the results of the baselines in the first section of the table, we observe (1) generally better performance of $\widehat{LM}$ in comparison to $\widehat{BM25}$ on the baselines across all evaluation metrics, and (2) only slight improvements of the `SkipGram` and `LSI` methods in comparison to `None`, with differences being significant mainly for MAP of the $\widehat{BM25}$ model.

The second section of Table 1 shows the effect of combining semantic lexicons with word similarities. Except for the case of PPDB on MAP of the $\widehat{LM}$ model, none of the semantic lexicon resources introduce significant improvements with respect to the baselines.

The third section shows the results of exploiting LSI as an external resource to combine with the Skipgram word embedding. The `PostFilter(LSI)` and `Retro(LSI)` both significantly improve all baselines. Specifically, `PostFilter(LSI)` performs better on MAP (precision-based), showing significant MAP improvements to baselines in both IR models. On the other hand, `Retro(LSI)` shows stronger performance on recall-based metrics by significantly improving the baselines on RECALL using $\widehat{LM}$.

We assume that the better performance of the `PostFilter(LSI)` method on MAP is due to its conservative approach, as the method only keeps those related words which are common in both Skipgram and LSI word embeddings. The `Retro(LSI)` method, however, incorporates LSI similarity in the vector representation space, providing wider semantic similarity scopes for words (useful for recall), while still maintaining MAP results in the same range as or higher than the baselines.



(a) Average Precision

(b) RECALL

Fig. 1: Per-query retrieval performance of the $\widehat{LM}$ model on the CLEF-IP 2013 using None, SkipGram, Retro(LSI) and PostFilter(LSI)

Finally, the results of the methods with two resources, namely LSI and PPDB (the best performance among the semantic lexicons), are shown in the last section of the table. Neither of the methods (Ext-Retro and PF-Retro) consistently outperform Retro(LSI) and PostFilter(LSI), suggesting that explicit semantic lexicons do not contribute to effectiveness improvements in this domain.

We continue our analysis by examining the robustness of the retrieval system when using the Retro(LSI) and PostFilter(LSI) word similarity methods. Figure 1 depicts Average Precision (AP) and RECALL per query using $\widehat{LM}$ (as it performs better in general and in particular for the baselines) with None, SkipGram (as used in [18,16]), Retro(LSI), and PostFilter(LSI) methods. We study the robustness of the compared methods by observing the consistency of the results across queries in comparison to the results of the None method (no word similarity information).

Turning to SkipGram, we observe cases of both improved and deteriorated results in comparison to None, indicating a lack of robustness of the method. Tracing the reason, similar to [17] we observe several cases of topic shifting, e.g., the query term *platinum* is expanded with *palladium* and *rhodium*, causing performance losses.

In contrast, PostFilter(LSI) shows highly robust performance, attaining the same or a better level of performance than None on almost all queries on both metrics. The same characteristic applies to Retro(LSI) on the RECALL metric, confirming the effectiveness as well as robustness of using the Retro(LSI) embedding on patent retrieval for the RECALL metric.

5 Conclusion

We study the effects of enriching word embeddings for patent retrieval using a global context. Observing considerable limitations when using local-context word embeddings (word2vec Skip-gram) in patent retrieval, we suggest incorporating additional information, obtained via LSI based on global contexts. We incorporate this additional source of information via retrofitting and post-filtering methods. Using our multi-context word embeddings, we observe significant improvements over the respective retrieval baselines on the CLEF-IP 2013 task. We report early results of an ongoing line of inquiry. In the future, we intend to explore the generality of our findings by investigating retrieval domains with similar characteristics to the patent retrieval setting.

References

1. L. Andersson, M. Lupu, J. Palotti, A. Hanbury, and A. Rauber. When is the time ripe for natural language processing for patent passage retrieval? In *Proc. of CIKM*, 2016.
2. C. F. Baker, C. J. Fillmore, and J. B. Lowe. The Berkeley FrameNet Project. In *Proc. of ACL*, 1998.
3. A. Berger and J. Lafferty. Information Retrieval As Statistical Translation. In *Proc. of SIGIR*, 1999.
4. F. Diaz, B. Mitra, and N. Craswell. Query Expansion with Locally-Trained Word Embeddings. In *In Proc. ACL*, 2016.
5. M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith. Retrofitting Word Vectors to Semantic Lexicons. In *In Proc. NAACL-HLT*, 2015.
6. J. Ganitkevitch, B. Van Durme, and C. Callison-Burch. PPDB: The Paraphrase Database. In *Proc. of NAACL*, 2013.
7. S. Kuzi, A. Shtok, and O. Kurland. Query expansion using word embeddings. In *Proc. of CIKM*, 2016.
8. M. Lupu. On the usability of random indexing in patent retrieval. In *Proc. of International Conference on Conceptual Structures*, 2014.
9. M. Lupu and A. Hanbury. Patent Retrieval. *Foundations and Trends in Information Retrieval*, 2013.
10. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Proc. of NIPS*, 2013.
11. G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 1995.
12. G.-H. Nguyen, L. Soulier, L. Tamine, and N. Bricon-Souf. Dsrim: A deep neural information retrieval model enhanced by a knowledge resource driven representation of documents. In *Proc. of SIGIR*, 2017.
13. F. Piroi, M. Lupu, and A. Hanbury. Overview of clef-ip 2013 lab. In *Proc. of CLEF*, 2013.
14. J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proc. of SIGIR*, 1998.
15. R. Řehůřek and P. Sojka. Software framework for topic modelling with large corpora. In *Proc. of LREC Workshop on New Challenges for NLP Frameworks*, 2010.
16. N. Rekabsaz, M. Lupu, and A. Hanbury. Exploration of a threshold for similarity based on uncertainty in word embedding. In *Proc. of ECIR*, 2017.
17. N. Rekabsaz, M. Lupu, A. Hanbury, and H. Zamani. Word embedding causes topic shifting; Exploit global context! In *Proc. of SIGIR*, 2017.
18. N. Rekabsaz, M. Lupu, A. Hanbury, and G. Zuccon. Generalizing Translation Models in the Probabilistic Relevance Framework. In *Proc. of CIKM*, 2016.
19. C. Xiong, J. Callan, and T.-Y. Liu. Word-entity duet representations for document ranking. In *Proc. of SIGIR*, 2017.
20. C. Xiong, Z. Dai, J. Callan, Z. Liu, and R. Power. End-to-end neural ad-hoc ranking with kernel pooling. In *Proc. of SIGIR*, 2017.
21. H. Zamani and W. B. Croft. Relevance-based word embedding. In *Proc. of SIGIR*, 2017.
22. G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proc. of Australasian Document Computing Symposium*, 2015.