PatentExplorer: Refining Patent Search with Domain-specific Topic Models

Mark Buckley
Siemens AG
Munich, Germany
mark.buckley@siemens.com

Sophia Althammer*
TU Vienna
Vienna, Austria
sophia.althammer@tuwien.ac.at

Arber Qoku*
German Cancer Consortium (DKTK)
German Cancer Research Center (DKFZ)
Heidelberg, Germany
arber.qoku@dkfz-heidelberg.de

ABSTRACT

Practitioners in the patent domain require high recall search solutions with precise results to be found in a large search space. Traditional search solutions focus on retrieving semantically similar documents, however we reason that the different topics in a patent document should be taken into account for search. In this paper we present PatentExplorer, an in-use system for patent search, which empowers users to explore different topics of semantically similar patents and refine the search by filtering by these topics. PatentExplorer uses similarity search to first retrieve patents for a list of patent IDs or given patent text and then offers the ability to refine the search results by their different topics using topic models trained on the domains in which our users are active.

KEYWORDS

Patent search, Topic models, User interface

1 INTRODUCTION

The ever-increasing volume [1] and linguistic complexity of published patent documents mean that searching for both high precision and high recall results for a given information need is a challenging problem. Practitioners in the patent domain require search results of high quality [21], as they provide the input to processes such as infringement litigation or freedom-to-operate clearing [15, 23]. The use of machine learning and deep learning methods for patent analysis is a vibrant research area [5, 12] with application in technology forecasting, patent retrieval [4, 19], patent text generation [13] or litigation analysis. There has been much research on the patent domain language which shows that the sections in patents constitute different genres depending on their legal or technical purpose [20, 23]. We reason that patents consist of different topics contained in the different sections of the document. The example in Figure 1 shows how a patent in the field of database systems can include topics such as physical storage of data or search interfaces—for a given patent search goal one of these could be relevant while the other is not. In industrial settings it is additionally important that search tools are particularly sensitive to individual companies' domains of interest, thereby improving the quality of search results.

> A real time database system configured to store database content…
> such that the replicas of each partition are contained on different physical storage units…
> wherein the system provides an interface for user searches for documents types including video, audio…

Figure 1: Example (abriged) of a multi-topic patent text

To provide an effective patent search tool under these conditions we present PatentExplorer, an in-use system for patent search, which empowers the users to explore different topics in search results and refine the results by their topics. PatentExplorer uses similarity search for first stage retrieval and domain-specific topic modelling for refinement of the search results. We propose topic modelling for search refinement because it is typical that a patent document will deal with multiple related but orthogonal subjects. For a particular information need, some but not all of these will be relevant. Therefore we combine a document level analysis (similarity) with a sub-document level analysis (topic models) for patent search. The intention is that the user can retrieve a large set of semantically related patents and inspect the topic distributions of the most similar ones. In order to refine the results the user can apply filters on specific topics, thereby increasing the task-specific relevance of the most highly ranked results.

This paper presents the design and user interface of the in-use web application which implements this idea as well as the technical description. The system has been designed with a particular user persona in mind. The intended user is a patent search professional, and therefore is familiar with patent search tools and also has deep knowledge of existing patent search methodologies, such as boolean retrieval and category filtering, as well as having broad technical knowledge of the relevant industrial domains.

2 BACKGROUND

In this section we give some background about related work on patent search tools, furthermore we introduce the methods for similarity search and topic models which we employ in PatentExplorer.

[*] Work done while at Siemens AG.

PatentSemTech, July 15th, 2021, online

2.1 Related work

Patent search holds several domain-specific challenges for information retrieval [15]. Furthermore serving the specific use-case setting of practitioners in a company requires company-specific adaptation of the search solution. Different techniques and approaches have been explored to improve and refine the search results in the patent domain, ranging from query expansion [2, 16, 25] to term selection [10]. For prior art retrieval in the CLEF-IP workshop [19], Verma and Varma [26] demonstrate high retrieval performance by representing a patent document by its IPC classes and computing similarity of patents based on the IPC classes. For patent search tools, mainly the challenge of high coverage of all published patents is addressed with an federated approach [22] or a single access point via text editor [7].

2.2 Methods

2.2.1 Similarity search. Similarity search is a method for retrieval where for a given query document, a ranked list of semantically relevant documents is computed, as shown in Figure 2. The general approach is to first embed the query document into a vector representation which encodes its semantics. This representation is then compared to the equivalent representations for each of the known documents in the search index. The results are then sorted by similarity score and the highest ranking results are presented to the user. The similarity function is usually cosine similarity.

The crucial step is to find an embedding which computes a suitable document representation. Different representations have been used in previous research, for instance tf-idf weighted sparse representations, latent semantic indexing, or contextualised document embeddings, for instance computed by a BERT model [9].

Despite the semantic richness of contextualised document embeddings, sparse representations have been found to be competitive in large scale retrieval scenarios [14]. We employ tf-idf weighted sparse representation in PatentExplorer for retrieving similar patents in the first stage. Large scale retrieval needs to use efficient indexing, such as algorithms for approximate nearest neighbour search [11], to avoid computing the cosine similarity scores for every document in the search space. Therefore we employ approximate nearest neighbor search on the sparse representations in PatentExplorer.

2.2.2 Topic models. Topic models help to understand the internal structure of large text data sets by summarising the themes which occur in the documents [8]. Topic modelling is an unsupervised approach (ie no labelled data is required) and can be applied to any domain. The only assumptions are the distributional hypothesis, that the frequency of occurrence of words and phrases is a good reflection of the strength and prevalence of themes, and the assumption that in general documents are a mixture of several topics. The topic modelling process begins by converting a set of documents into a sparse term-document matrix T containing weighted feature frequencies for each document. The topic modelling algorithm transforms this matrix into a pair of matrices Z and D such that

$$T \approx Z \times D$$

Z, the term-topic matrix, encodes the weight of each feature with respect to the topics and D, the document-topic matrix, contains a latent representation for each document showing which topics it belongs to.

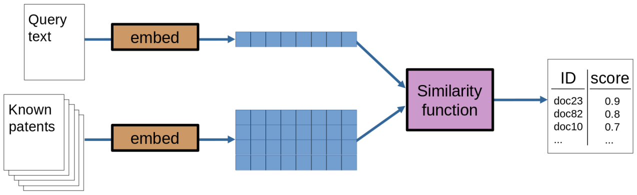
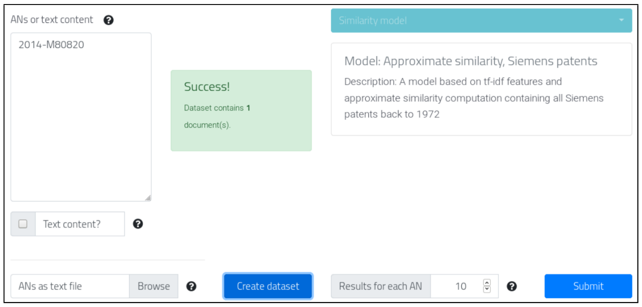


Figure 2: Similarity search process



2014-M80820

Success!

Dataset contains **1** document(s).

Model: Approximate similarity, Siemens patents

Description: A model based on tf-idf features and approximate similarity computation containing all Siemens patents back to 1972

☐ Text content?

Results for each AN 10

Figure 3: Similarity Search in PatentExplorer entering a list of patent IDs or a patent text

We consider two algorithms for topic modelling in this work, latent Dirichlet allocation (LDA) [6] and non-negative matrix factorisation (NMF) [24]. LDA is a generative model which treats documents as a distribution over topics and topics as a distribution over words. NMF is a method for decomposing large matrices of non-negative values into the product of smaller matrices, in this case into the matrices Z and D. In each case the topic distributions (the rows of the matrix D) can be interpreted as a document representation and thus can be compared and analysed. The index of the largest value of each row of D is interpreted as the most likely topic for that document. Topic modelling has previously been used in the patent domain, for instance for technology forecasting [23].

3 PATENTEXPLORER

In this section we first show the user interface of PatentExplorer and give some implementation details about the architecture, the data and the similarity and topic models being employed in PatentExplorer.

3.1 User interface

The user interaction begins with the submission of a list of patent IDs (accession numbers) or the text of a patent, as shown in Figure 3. The system retrieves the text of the patents given in the list of patent IDs and creates a local copy of the text content of each of the patents. How many of the patents in the list are found in the index is indicated with "Dataset contains - documents". The user can then submit the "Dataset" to the system to retrieve similar documents based on the similarity search.

For each of the similar documents, the system also computes their topic distribution. The distribution is displayed along with the accession number and similarity score between the query patent and each similar patent, as shown in Figure 4. The most highly weighted words for each topic, drawn from the matrix Z, are displayed by hovering over the bars. The figure also shows the filter function which the system provides to re-rank the search results according to their topics. Both positive and negative filters can be applied. Positive filters lead to matching documents being lifted to the top of the ranked list, negative filters lead to the matching documents being discarded from the results set. For both filter types, a list of topics can be specified in the text field on the left hand side, as well as two slider values. The two slider values restrict when the filter will match: a document matches if at least one of the chosen topics has a weight in the topic distribution of that document of at least "min probability". The default value is 0.1. With a max rank of r, the filter will also only match if the chosen topic is among the r most highly weighted topics in the distribution for that document. So if the query document is the example in Figure 1, the user could inspect the topic distribution to find, for instance, the topic concerning physical storage, and apply a negative filter to remove it, leaving those results which have more to do with user search. Finally, when the user is finished applying filters to the search results, the results set can be downloaded as tabular data, preserving the filtered order and including similarity scores.

3.2 Technical implementation

3.2.1 Data. To prepare the components of our system we collected two overlapping data sets. The source is a commercially provided database of patent abstracts in which patents from patent offices worldwide have been translated into a consistent, English-language form. We chose this data source in order to achieve maximum uniformity of the input data, however PatentExplorer makes no strong assumptions about the content of the documents, and would also work on publicly available patent data. The *Our-Portfolio* data set contains the patents whose assignee is our company or its subsidiaries. It contains 73k documents. We filtered this data set to only contain patents filed since 2010, resulting in a set of 36k documents. The *All-Patents* data set is the collection of all patents published between 2014 and 2020, which contains approximately 15 million documents. For both data sets we extract the title and abstract of the patents.

3.2.2 Architecture. The architecture of the system is shown in Figure 5. The two main components are the similarity search and the topic model. Each component offers an API with one function: "get-similar-ids" and "get-topic-distribution", respectively. The "get-similar-ids" function receives one or more patent IDs and retrieves the most similar documents from the search index, defined as the cosine similarity between their representations. This is equivalent to finding the nearest neighbours of the query document in the representation space. The "get-topic-distribution" receives a single patent ID and computes the topic distribution for that document from the previously trained topic model. The search index and the topic model are static resources which are not changed during run time. Both components retrieve the patent document content from the database "Patent documents" directly as required, so that the user must only supply document IDs.

	$50k$	$100k$	$250k$
NMF	0.65	0.67	0.69
LDA	0.61	0.63	0.65

Table 1: Coherence scores (C_{NPMI}) for NMF and LDA across three data set sizes. Each score is the average over the coherence scores for $k \in \{5, 10, ..., 95, 100\}$

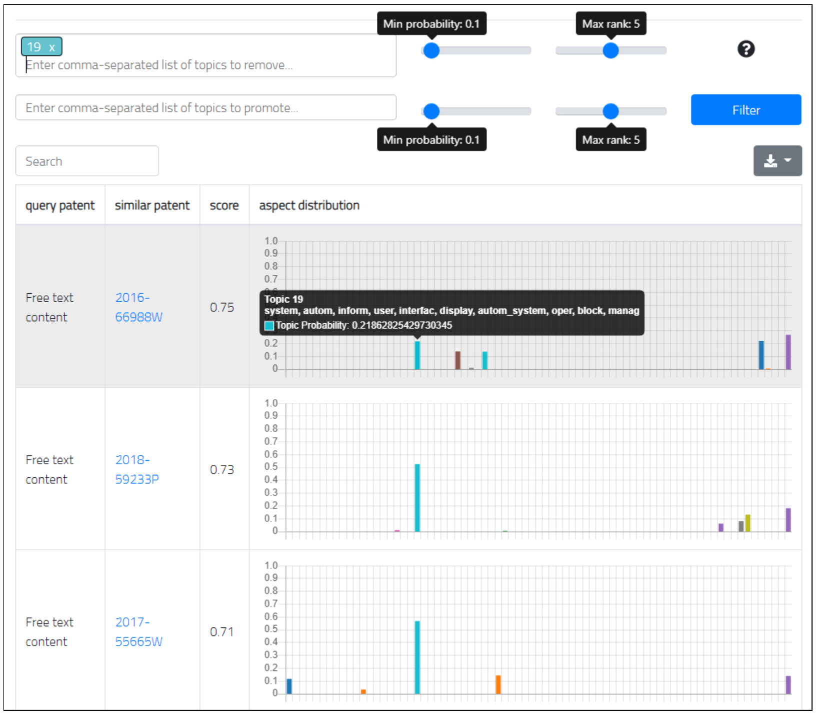
3.2.3 Training the topic model. The topic model is trained on the *Our-Portfolio* data set. The documents were preprocessed to remove approximately 50 patent-specific stop words, such as "invention" or "apparatus", as well as usual English stop words. We performed stemming and then extracted all n-grams for $n = 1, 2, 3, 4$ to construct the term-document matrix. We discarded words which occurred in fewer than 10 documents or in more than 40% of the documents.

In preliminary experiments we used a coherence metric to investigate the optimal parameters for the topic model. In recent years, several approaches to measure coherence have been developed based on distributional properties of word pairs over a set of words [17, 18], which mostly differ in the pairwise scoring metric being used. A typical choice is pointwise mutual information (PMI), which measures the strength of association between words in a data set within windows of a given size.

We use the coherence score C_{NPMI} as proposed by Aletras and Stevenson [3]. An N-dimensional context vector is created for each word w, whose elements are the normalised PMI values of w with each of the other top words of the topic. Each word w is then assigned the cosine similarity of its context vector and the sum of the other context vectors. The coherence score of the topic is the average of all of these cosine similarities.

To investigate which parametrisation of topic modelling works best for patent text we took a sample of 513k English-language patents from those published in 2010. We removed duplicates and documents which were either very long or very short, leaving a set of approximately 255k documents. As we show in Table 1, both LDA and NMF exhibit similar performance on this data set, as measured by C_{NPMI}, with NMF discovering marginally better topics. We find upon manual inspection that NMF is more robust across a wide range of number of topics. We therefore choose NMF to implement the system. We finally use NMF with 75 topics to train the topic model for the system on the *Our-Portfolio* data set.

3.2.4 Compiling the search index. To compile the search index we must first compute an embedding for each document in the search space. We use latent semantic indexing (LSI) to compute the document vectors, which is the result of tf-idf vectorisation followed by SVD compression [8]. Rather than computing the tf-idf weights from the entire *All-Patents* data set, we instead compute the tf-idf weights from the *Our-Portfolio* data set, so that each document embedding in the search space will encode information which is relevant to our industrial domains. We then apply an SVD compression into 200 dimensions in order to reduce the size of each document vector and therefore the size of the overall search index. We use the resulting LSI projection function to compute a document embedding for each of the 15m documents in the *All-Patents* data set.



19 x

Enter comma-separated list of topics to remove...

Min probability: 0.1

Max rank: 5

Enter comma-separated list of topics to promote...

Min probability: 0.1

Max rank: 5

Filter

Search

query patent	similar patent	score	aspect distribution
Free text content	2016-66988W	0.75	Topic 19 system, autom, inform, user, interfac, display, autom_system, oper, block, manag Topic Probability: 0.21862825429730345
Free text content	2018-59233P	0.73	
Free text content	2017-55665W	0.71	

Figure 4: PatentExplorer interface for exploring and refining the topic distribution of the search results.

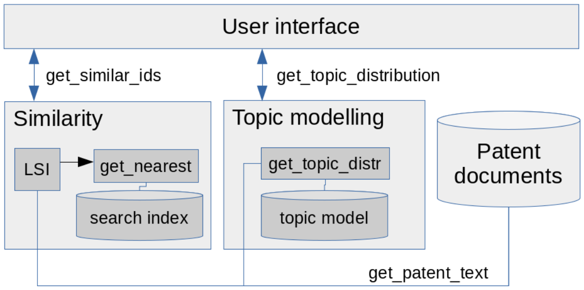


Figure 5: System architecture of PatentExplorer containing the Similarity Search and Topic Modelling component

To implement the lookup of documents given a query document we use Annoy[1], a library which provides approximate nearest neighbour search. Each document embedding is normalised before insertion so that the cosine similarity can be computed with the dot product function. The similarity component of the system provides an endpoint which returns the IDs of the n most similar documents for some query document and some n.

4 CONCLUSION AND FUTURE WORK

In this paper we present PatentExplorer, an in-use system for patent search. PatentExplorer gives users the ability to retrieve similar patents given a list of patent IDs or the patent text and refine their search results depending on the different topics of the patents. The topic models are tailored to the domain-specific topics of a company operating in the technical domain.

Tailoring the search representation and topic models to our domains turned out in initial user testing to offer mixed results. Feedback from patent search experts indicates that while the system can deliver relevant results within our domains, outside of these domains it can return results with few or no relevant documents among the ten highest ranked results. While building and testing our system we have found that the requirements of patent search use cases place high demands on the accuracy of dedicated search tools. In order to reduce the latency of the similarity search to an acceptable level we were forced to simplify the similarity computation, using a compressed tf-idf representation where a contextualised document embedding may well have produced better results. It is also crucial to provide full coverage: The dataset of patents which the system contains goes back to 2014, however for prior art searches, all previously published patents should be discoverable. Finally the need to update the search index continuously leads to considerable recurring computational load and data management tasks—this is not yet provided for.

Our future work to improve the system will include expanding the system architecture to efficiently handle a larger number of documents in the search space. In the longer term we intend to investigate introducing more appropriate document representations to be used in the search index, for instance by using a large language model such as BERT, or by learning the representations via a supervised auxiliary task.

[1] https://github.com/spotify/annoy

REFERENCES

[1] [n.d.]. U.S. Patent Statistics Chart. https://www.uspto.gov/web/offices/ac/ido/oeip/taf/us_stat.htm. Accessed: 2021-06-04.

[2] Bashar Al-Shboul and Sung-Hyon Myaeng. 2011. Query Phrase Expansion Using Wikipedia in Patent Class Search. In *Information Retrieval Technology*, Mohamed Vall Mohamed Salem, Khaled Shaalan, Farhad Oroumchian, Azadeh Shakery, and Halim Khelalfa (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 115–126.

[3] Nikolaos Aletras and Mark Stevenson. 2013. Evaluating Topic Coherence Using Distributional Semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*. Association for Computational Linguistics, Potsdam, Germany, 13–22. https://www.aclweb.org/anthology/W13-0102

[4] Sophia Althammer, Sebastian Hofstätter, and Allan Hanbury. 2021. Cross-domain Retrieval in the Legal and Patent Domains: a Reproducibility Study. In *Advances in Information Retrieval, 43rd European Conference on IR Research, ECIR 2021*.

[5] Leonidas Aristodemou and Frank Tietze. 2018. The state-of-the-art on Intellectual Property Analytics (IPA): A literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (IP) data. *World Patent Information* 55 (12 2018), 37–51. https://doi.org/10.1016/j.wpi.2018.07.002

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[7] Manajit Chakraborty, David Zimmermann, and Fabio Crestani. 2021. PatentQuest: A User-Oriented Tool for Integrated Patent Search. In *Proceedings of the 11th International Workshop on Bibliometric-enhanced Information Retrieval co-located with 43rd European Conference on Information Retrieval (ECIR 2021), Lucca, Italy (online only), April 1st, 2021 (CEUR Workshop Proceedings, Vol. 2847)*, Ingo Frommholz, Philipp Mayr, Guillaume Cabanac, and Suzan Verberne (Eds.). CEUR-WS.org, 89–101. http://ceur-ws.org/Vol-2847/paper-09.pdf

[8] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41, 6 (1990), 391–407.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[10] Mona Golestan Far, Scott Sanner, Mohamed Reda Bouadjenek, Gabriela Ferraro, and David Hawking. 2015. On Term Selection Techniques for Patent Prior Art Search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Santiago, Chile) *(SIGIR '15)*. Association for Computing Machinery, New York, NY, USA, 803–806. https://doi.org/10.1145/2766462.2767801

[11] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* (2019), 1–1. https://doi.org/10.1109/TBDATA.2019.2921572

[12] Ralf Krestel, Renukswamy Chikkamath, Christoph Hewel, and Julian Risch. 2021. A survey on deep learning for patent analysis. *World Patent Information* 65 (6 2021). https://doi.org/10.1016/j.wpi.2021.102035

[13] Jieh-Sheng Lee and Jieh Hsiang. 2020. PatentTransformer-2: Controlling Patent Text Generation by Structural Metadata. arXiv:2001.03708 [cs.CL]

[14] Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2020. Sparse, Dense, and Attentional Representations for Text Retrieval. *CoRR* abs/2005.00181 (2020). arXiv:2005.00181 https://arxiv.org/abs/2005.00181

[15] Mihai Lupu and Allan Hanbury. 2013. Patent Retrieval. *Foundations and Trends® in Information Retrieval* 7, 1 (2013), 1–97. https://doi.org/10.1561/1500000027

[16] Walid Magdy and Gareth J.F. Jones. 2011. A Study on Query Expansion Methods for Patent Retrieval. In *Proceedings of the 4th Workshop on Patent Information Retrieval* (Glasgow, Scotland, UK) *(PaIR '11)*. Association for Computing Machinery, New York, NY, USA, 19–24. https://doi.org/10.1145/2064975.2064982

[17] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 262–272.

[18] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 100–108.

[19] Florina Piroi, Mihai Lupu, and Allan Hanbury. 2013. Overview of CLEF-IP 2013 Lab. In *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, Pamela Forner, Henning Müller, Roberto Paredes, Paolo Rosso, and Benno Stein (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 232–249.

[20] Julian Risch and Ralf Krestel. 2019. Domain-specific word embeddings for patent classification. *Data Technol. Appl.* 53 (2019), 108–122.

[21] Tony Russell-Rose, Jon Chamberlain, and Leif Azzopardi. 2018. Information retrieval in the workplace: A comparison of professional search practices. *Information Processing and Management* 54 (11 2018), 1042–1057. Issue 6. https://doi.org/10.1016/j.ipm.2018.07.003

[22] Mike Salampasis and Allan Hanbury. 2014. PerFedPat: An integrated federated system for patent search. *World Patent Information* 38 (09 2014). https://doi.org/10.1016/j.wpi.2014.08.001

[23] Walid Shalaby and Wlodek Zadrozny. 2019. Patent retrieval : a literature review. *Knowledge and Information Systems* (2019). https://doi.org/10.1007/s10115-018-1322-7

[24] Rashish Tandon and Suvrit Sra. 2010. Sparse nonnegative matrix approximation: new formulations and algorithms. (2010).

[25] Wolfgang Tannebaum, Parvaz Mahdabi, and Andreas Rauber. 2015. Effect of Log-Based Query Term Expansion on Retrieval Effectiveness in Patent Searching, Vol. 9283. 300–305. https://doi.org/10.1007/978-3-319-24027-5_32

[26] Manisha Verma and Vasudeva Varma. 2011. Exploring Keyphrase Extraction and IPC Classification Vectors for Prior Art Search., Vol. 1177.