Journal of Physics: Conference Series

Fast and Accurate Patent Classification in Search Engines

Fast and Accurate Patent Classification in Search Engines

Vasiliy Yadrintsev[1, 2], Amir Bakarov[1, 3], Roman Suvorov[1] and Ilya Sochenkov[1, 4]

[1] Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, Moscow, Russia
[2] Peoples Friendship University of Russia (RUDN University), Moscow, Russia
[3] The National Research University Higher School of Economics, Moscow, Russia
[4] Skolkovo Institute of Science and Technology, Moscow, Russia

E-mail: vvyadrincev@gmail.com

Abstract. This article presents a new approach to large scale patent classification. The need to classify documents often takes place in professional information retrieval systems. In this paper we describe our approach, based on linguistically-supported k-nearest neighbors. We experimentally evaluate it on the Russian and English datasets and compare modern classification technique fastText. We show that KNN is a viable alternative to traditional text classifiers, achieving comparable accuracy while using less additional hardware resources.

1. Introduction

Due to the rapid development of information technology and the increase in the number of electronic documents, such as patents, it is impossible to manually process and manage large amounts of text documents. The most natural way to search almost any document collection is by using keywords [1]. Unfortunately, there are some obstacles to the sole use of keywords for searching patents [1]: extremely complicated patent language and etc. The use of classification information also does not solve the problem. In particular, the patent classification system is quite complicated, and then there is a problem of determining the subject matter of a patent by human experts. It is a common practice fully studying the text of the patent document for determining the correct class, for example from the International Patent Classification (IPC) system. Thus, the automated assignment of classes to patents is an important issue, especially for patent offices. The task of automatic patent classification is well known. However, the current level of solving this problem, in particular recall and precision, does not correspond to industrial application.

The purpose of this work is the investigation of the use of the vector space models extended by nominal groups for the classification of patent documents in search engines. We carried out experiments on several levels of the IPC on Russian and English patents. There is the task of supervised learning – multi-label and multi-class classification of text documents. We are not aware for any studies that try to exploit the extension by nominal groups, so we consider this feature of our work as our major contribution. Our work is also among the first studies on the patent classification task for Russian language.

This study is organized as follows. Section 2 describes the methodology of our approach to patent classification. Section 3 gives more details about our experimental setup. Section 4 proposes results and a discussion on them. Section 5 puts out work in the context of previous studies, describing other approaches to the patent classification task. Section 6 concludes the article.

2. The Retrieval Approach for Patent Classification

A method of searching for topically similar documents is used, which is based on a comparison of vectors of lexical descriptors [2]. The method of searching for topically similar documents accepts text document (in our case, text of the patent – abstract, description, claims, meta- information) as input, then it formed ranked by thematic proximities list of similar documents from beforehand indexed collection of documents. Search for thematically similar documents can also be used to identify scientific directions using metric thematic clustering [3].

The use of the search engine for similarity search assumes that the first step is indexing [4]. Since there is vector space model (Bag-of-Words) [5], you cannot index training documents and test documents into one collection. Thus, it is necessary beforehand to divide the sample to training and testing parts.

3. Experimental Setup

We compared our approach with fastText [6] and the best classifier of CLEF-IP competition [7]. In fastText classification algorithm, a sentence (or a document) vector is obtained by averaging the n-gram embeddings, and then a multinomial logistic regression exploits these vectors as a features.

Full texts of patent documents on different languages (English and Russian) are used. The data is split randomly three times on the training and test parts in a ratio of 70 to 30. Three times independently, we trained classifiers and evaluated. Then we averaged the results. We did not use the cross-validation, because with this approach training sets most likely have “poor” classes (classes whose representative documents are small in the set).

As classes for classification independently were used three levels of the IPC: subclasses, maingroups, and subgroups. A specific version of the IPC was not used in this work: a set of classification labels was compiled from the available data set. In this case, two identical by the name of the label, but from two different versions of the IPC (for example: *H01L21/70 2006.1* and *H01L21/70 2009.1*) were considered as one class *H01L21/70*.

As table 1 shows, for CLEF there are approximately 630 labels for subclass level, approximately 6800 labels for maingroup level and more than 57000 for subgroup level. For FIPS there are approximately 650 labels for subclass level, approximately 7220 labels for maingroup level and about 50500 for subgroup level.

Table 1. Datasets.

Collection	Description	Documents	IPC labels by level
CLEF Eng	English subset	699 000	subclass: 630 maingroup: 6800 subgroup: 57000
CLEF Eng Abs/Desc	English subset, that contains abstract or description	531 000	about the same as above
FIPS	Russian Federal Institute of Industrial Property(FIPS)	776 000	subclass: 650 maingroup: 7220 subgroup: 50500

We evaluated classifiers by the well-known metrics: f1-measure, precision, recall with micro and macro averaging. Also we evaluated our classifier on the CLEF-IP test sample (1000 documents) for English. Thus, the following experiments were carried out:

- Splitting data in a ratio of 70 to 30;
- “CLEF Eng” as train data and CLEF-IP test sample.

Similarity search parameters optimization
To optimize the similarity search parameters, we solved as a typical minimax problem – to maximize the difference between the minimum of interclass distance and the maximum of intraclass distance. Also possible the use of the average, median and another functions instead of pure minimum or maximum functions.

We selected special subset of classes and documents for optimizing: for each section three most common classes, that have more than 1000 representative documents; test subset – 300 random docs for each selected class. There are 8 sections in IPC: A, B, C, D, E, F, G, H the first symbols of IPC classes. For example: for the **H** section for subgroup-level optimizing we selected classes **H**01L21/02(11 452 documents with this label in CLEF collection), **H**01L21/70(5 209 in CLEF collection) and **H**01L29/66(3 972 in CLEF collection). Also we selected 3 classes for the sections A, C, G and 2 classes for B. We assume that the set of documents chosen in this way is sufficient for optimization.

And also we reduced the size of words in vector space model by removing rare words and phrases, we tried three options:
- without deleting,
- deleting word or phrase that occurs less than 3 times,
- ...less than 5 times.

4. Results and Discussion

Table 2 shows the main results for subclass level for our (KNN) method. Without reducing results are lower than in the table the difference is slightly more than 1 percent. Table 2 shows that there is no difference between 'reduce 3' and 'reduce 5'. We optimized parameters of similarity search for the subgroup level and for the subclass level. Difference between the subgroup optimizing and subclass optimizing is that in first situation we use subclass labels (the third level of IPC), in second subgroup (the fifth, last level). At first we thought that optimization at the lowest level would be enough, but the experiment showed that for classifying at the subclass level it is better to optimize at the subclass level. As shown in the table 2, subgroup optimizing without reducing for CLEF English subset showed 0.67 f1-measure and subclass optimizing with 'reduce 3' showed 0.7102. As noted above, reducing raises f1-measure by about 1 percent. Thus, optimization at the subclass level gives a better f1 of about 3 percent compared to optimization at the subgroup level (for CLEF English collection).

Table 2. Subclass, KNN. f1, precision, recall for best f1.

			Macro			**Micro**		**Optimized for,**
Collection	**Data**	**f1**	**prec.**	**rec.**	**f1**	**prec.**	**rec.**	**reduce**
CLEF Eng	70/30	**0.6434**	0.699	0.595	**0.7102**	0.738	0.683	subclass, 3
CLEF Eng	70/30	**0.614**	0.776	0.8306	**0.67**	0.707	0.854	subclass, no
FIPS	70/30	**0.712**	0.71	0.76	**0.696**	0.812	0.732	subgroup, no
CLEF Eng Abs/Desc	test 1000	**0.639**	0.666	0.614	**0.7098**	0.718	0.701	subclass, 3
CLEF Eng Abs/Desc	test 1000	**0.639**	0.666	0.614	**0.7098**	0.718	0.701	subclass, 5
CLEF Eng	test 1000	**0.639**	0.666	0.614	**0.7098**	0.718	0.701	subclass, 3
CLEF Eng	test 1000	**0.639**	0.666	0.614	**0.7098**	0.718	0.701	subclass, 5

Table 3 shows the bests of f1, precision, and recall for subclass level for KNN. For 70/30 splitting:
- With micro averaging the best f1 is 0.7102, best precision – 0.7644 and best recall – 0.857.
- With macro averaging the best f1 is 0.6434, best precision – 0.7909 and best recall – 0.886. KNN on CLEF-IP test sample:

- With micro averaging the best f1 is 0.7098, best precision – 0.755 and best recall – 0.9.
- With macro averaging the best f1 is 0.6393, best precision – 0.888 and best recall – 0.869.

Table 3. Subclass, CLEF, KNN. Best f1, precision, recall.

Collection	Data	f1	Macro prec.	rec.	f1	Micro prec.	rec.	Comment
CLEF Eng	70/30	**0.6434**	0.699	0.595	**0.7102**	0.738	0.683	best f1
CLEF Eng	70/30	0.605	**0.7909**	0.49	0.674	**0.7644**	0.603	best precision
CLEF Eng	70/30	0.404	0.262	**0.886**	0.352	0.221	**0.857**	best recall
CLEF Eng	test 1000	**0.6393**	0.666	0.6145	**0.7098**	0.7182	0.7015	best f1
CLEF Eng	test 1000	0.119	**0.888**	0.064	0.682	**0.755**	0.622	best precision
CLEF Eng	test 1000	0.289	0.173	**0.869**	0.303	0.182	**0.9**	best recall

Table 4 shows the bests of f1, precision, and recall for subclass level for fastText. For 70/30 splitting:

- With micro averaging the best f1 is 0.7098, best precision – 0.755 and best recall – 0.9.
- With macro averaging the best f1 is 0.6393, best precision – 0.888 and best recall – 0.869. fastText CLEF-IP test sample:
- With micro averaging the best f1 is 0.704, best precision – 0.747 and best recall – 0.684.
- With macro averaging the best f1 is 0.626, best precision – 0.865 and best recall – 0.601.

Table 4. Subclass, CLEF, fastText. Best f1, precision, recall.

Collection	Data	f1	Macro prec.	rec.	f1	Micro prec.	rec.	Comment
CLEF Eng	70/30	**0.6431**	0.696	0.597	**0.7305**	0.768	0.696	best f1
CLEF Eng	70/30	0.493	**0.887**	0.342	0.701	**0.8029**	0.622	best precision
CLEF Eng	70/30	0.637	0.668	**0.608**	0.728	0.763	**0.696**	best recall
CLEF Eng	test 1000	**0.626**	0.654	0.601	**0.704**	0.726	0.683	best f1
CLEF Eng	test 1000	0.502	**0.865**	0.354	0.673	**0.747**	0.613	best precision
CLEF Eng	test 1000	0.626	0.654	**0.601**	0.703	0.724	**0.684**	best recall

The best classifier in CLEF-IP competition has comparable results with our classifier results (micro): best f1 – 0.7059, best precision – 0.7443, best recall – 0.8657. Thus, with micro averaging the best f1 and precision showed a fastText – higher by more than 2% f1 than others, higher about 4% precision, and smaller about 14% recall. Our classifier and fastText showed the same f1 with macro averaging, but there is difference in precision and recall: fastText showed better precision, but worse recall.

Table 5. Subgroup, CLEF, fastText and KNN. f1, precision, recall for best f1.

Collection	Data	f1	Macro prec.	rec.	f1	Micro prec.	rec.	Method
CLEF Eng	70%/104th.	**0.398**	0.493	0.333	**0.431**	0.483	0.388	fastText
CLEF Eng	70%/104th.	**0.396**	0.428	0.369	**0.404**	0.394	0.414	KNN

Table 5 shows the main result on subgroup level for both our and fastText classifier. The training set is used the same as for the subclass level, but there is difference in test data sampling for subgroup level: only those labels were selected for which there are at least 20 documents in the training set, we also limited the number of documents for each label in test set – no more than five hundred.

The described restrictions allowed to select about 104 thousand documents from more than 200 thousand. Table 5 shows that there is no big difference between macro and micro averaging, because we restricted the number of test documents. Also from the table 5 it is seen that fastText has approximately 3% higher f1-measure with micro averaging, but with macro averaging f1-measure is the same for both KNN and fastText.

5. Related Work

The task of patent classification is well-known to text analytics researchers and industry companies. The first works on this topic were conducted by US patent offices [8–10]. Early studies were exploiting statistical natural language processing methods. The classification methodology of the later ones was based on extraction of keywords from patent documents with the help of a neural network [11]. Such works demonstrated decent results despite the evaluation was done on a relatively small test set.

In last few years new test sets were released as a part of CLEF-IP 2011 competition [7]. They were based on the data of such patent offices like European Patent Office (EPO) and World Intellectual Property Organization (WIPO). Recent studies exploiting this data have used more advanced features of patent documents like references and patent metadata [12]. They showed that the use of full descriptions of patent documents improves the quality of classification, reaching F1-score of 0.7 for the subclass level in the best run. Some works also tried to exploit this feature, proposing more accurate document representations for patent classification [13], or involving active learning methodologies [14]. The use of patent metadata was applied to other studies, for example, in several works the task of measurement the patent similarity was done with the help of comparison of patent titles and annotations [15].

6. Conclusion

In the proposed study we have investigated the use of the vector space models extended by nominal groups for the classification of patent documents in search engines. We carried out our experiments on the test sets of CLEF-2011 and FIPS on several patent classification levels for Russian and English languages.

Our system showed the best performance on the compared test sets and demonstrated that the use of nominal groups for the classification of patent documents in search engines could rapidly increase the quality of the classification.

In future we want to continue our research on this topic, extending it to other languages and more test data. We also plan to try to investigate more advanced approaches (for instance, the ELMo embeddings [16]) for obtaining text features.

Acknowledgments

This study was supported by the Ministry of Education and Science of the Russian Federation (grant 14.756.31.0001), by the "RUDN University Program 5-100", and partially funded by Russian Foundation for Basic Research according to the research project No. 16-29-12929

References

[1] Eisinger D, Tsatsaronis G, Bundschus M, Wieneke U and Schroeder M 2013 Automated patent categorization and guided patent search using ipc as inspired by mesh and pubmed *Journal of biomedical semantics* vol 4 (BioMed Central) p S3

[2] Sochenkov I V, Zubarev D V and Tikhomirov I A 2018 *Informatika i Ee Primeneniya [Informatics and its Applications]* **12** 89–94

[3] Shvets A, Devyatkin D, Sochenkov I, Tikhomirov I, Popov K and Yarygin K 2015 Detection of current research directions based on full-text clustering 2015 *Science and Information Conference* (SAI) 483–88

[4] Schütze H, Manning C D and Raghavan P 2008 *Introduction to information retrieval* vol 39 (Cambridge University Press)

[5] Manning C D, Manning C D and Schu¨tze H 1999 *Foundations of statistical natural language processing* (MIT press)

[6] Bojanowski P, Grave E, Joulin A and Mikolov T 2017 *Transactions of the Association of Computational Linguistics* **5** 135–46

[7] Piroi F, Lupu M, Hanbury A, Sexton A P, Magdy W and Filippov I V 2010 Clef-ip 2010: Retrieval experiments in the intellectual property domain. *CLEF (notebook papers/labs/workshops)*

[8] Krier M and Zacca F 2002 *World Patent Information* **24** 187–96

[9] Fall C J and Benzineb K 2002 *World Intellectual Property Organization* **29**

[10] Fall C J, T¨orcsv´ari A, Benzineb K and Karetka G 2003 Automated categorization in the international patent classification *Acm Sigir Forum* vol 37 (ACM) 10–25

[11] Trappey A J, Hsu F C, Trappey C V and Lin C I 2006 *Expert Systems with Applications* **31** 755–65

[12] Verberne S and D'hondt E 2011 Patent classification experiments with the linguistic classification system lcs in clef-ip 2011. *CLEF (Notebook Papers/Labs/Workshop)*

[13] D'hondt E, Verberne S, Koster C and Boves L 2013 *Computational Linguistics* **39** 755–75

[14] Zhang X 2014 *Neurocomputing* **127** 200–5

[15] Arts S, Cassiman B and Gomez J C 2018 *Strategic Management Journal* **39** 62–84

[16] Peters M, Neumann M, Iyyer M, Gardner M, Clark C, Lee K and Zettlemoyer L 2018 Deep contextualized word representations *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* vol 1 2227–37