Automatic categorization of patent applications using classifier combinations

Mathiassen, Henrik; Ortiz-Arroyo, Daniel

Published in:
Intelligent Data Engineering and Automated Learning - Ideal 2006, Proceedings

Publication date:
2006

Document Version
Early version, also known as pre-print

Link to publication from Aalborg University

Citation for published version (APA):
Mathiassen, H., & Ortiz-Arroyo, D. (2006). Automatic categorization of patent applications using classifier combinations. In *Intelligent Data Engineering and Automated Learning - Ideal 2006, Proceedings* (pp. 1039-1047)

Automatic Categorization of Patent Applications Using Classifier Combinations

Henrik Mathiassen[1] and Daniel Ortiz-Arroyo[2]

Computer Science and Engineering Department
Aalborg University, Esbjerg
Niels Bohrs Vej 8, 6700 Esbjerg Denmark
[1]hm1464@student.cs.aaue.dk, [2]do@cs.aaue.dk

Abstract. In this paper we explore the effectiveness of combining several machine learning based methods to categorize patent applications. Classifiers are constructed from each categorization method in the combination, based on the document representations where the best performance was obtained. Therefore, the ensemble of methods makes categorization predictions with knowledge observed from different perspectives. In addition, we explore the application of diverse combination techniques of classifiers to improve the overall performance of the ensemble. In our experiments a refined version of the WIPO-alpha[1] document collection was used to train and evaluate the classifiers. The combination ensemble that achieved the best performance obtained an improvement of 6.51% compared to the best performing classifier participating in the combination.

Keywords: Categorization, Machine Learning, Knowledge Management.

1 Introduction

A patent is a contract between the state and the applicant by which a temporary monopoly is granted in return for disclosing all details of an invention. Patent rights must be applied for at a patent office to gain rights in a country. Patent *classification schemes* are a hierarchical system of categories used to organize and index the technical content of patents so that a specific topic or area of technology can be identified easily and accurately. Different classification schemes are used in the different patent organizations. The most widely used classification scheme is the *International Patent Classification (IPC)*. The IPC is a hierarchical categorization system comprising sections, classes, subclasses and groups (main groups and subgroups). The eighth edition of the IPC contains approximately 70,000 groups. Every subdivision of the IPC is indicated by a symbol and has a title. The IPC divides all technological fields into eight sections designated by one of the capital letters A through H. The sections include from *Human Necessities* and *Physics* to *Electricity, Textiles* and *Mechanics* among others. Each section, in turn, is subdivided into classes

[1] WIPO-alpha document collection available at http://www.wipo.int/ibis/datasets/index.html

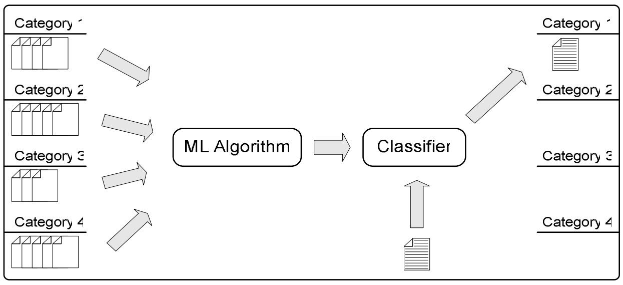

Fig. 1. Automated patent categorization based on machine learning

labeled with a section symbol followed by a two-digit number. Each class then contains one or several subclasses labeled with a class id followed by a capital letter, e.g. A01B. Finally, each subclass is broken down into subdivisions referred to as groups and known as either main groups or subgroups. The IPC is developed and administered by *World Intellectual Property Organization (WIPO)*. The WIPO-alpha collection, a publicly available dataset aimed at encouraging research in automated categorization of patent documents, contains 75,000 patent documents in English divided into a training set of 46,324 documents and a test set with 28,926 documents.

An intellectually built (i.e. human made) taxonomy is the only solution when patent categories are new and empty. However, since normally a great amount of manually categorized patent examples exist, it is feasible to apply *machine learning* techniques. The machine learning techniques employed in patent categorization are normally based on *supervised* learning. In *supervised* learning some examples called training documents are assigned to the correct category first. Then, based on the learned information from these examples, new unseen documents are categorized. Fig. 1 shows the general scheme of an automated patent categorization method. The module used to categorize documents is called the *classifier*. The classifier is trained using machine learning algorithms from an inductive process called the training/learning phase.

Patents are normally processed within organizations in two main stages: *pre-categorization* where it is determined the technical unit that will handle a patent and the *categorization* stage where the final category is assigned.

In this paper we explore the effectiveness of applying diverse techniques for combining supervised machine learning methods to automatically categorize patents. The techniques presented in this paper are aimed to automate the pre-categorization stage of patents. The rest of the paper is organized as follows. Section 2 discusses related research on patent categorization. Section 3 describes our proposed model for a patent categorization system. Experimental results of our model are presented in Section 4. Finally, Section 5 describes future work and presents some conclusions.

2 Related Work

The first reported research on patent categorization is the work by Chakrabarti et al. In [2] they propose a statistical model that attempts to categorize patents into a hierarchical model containing 3 categories subdivided into 12 subcategories. The classifier obtained a precision of 64% when was applied to patents. The authors argue that this relatively poor performance is caused by the diversities in authorship performed across time and assignees. To improve performance they attempted to use information contained in links between referencing patents. Naively indexing features from referenced patents is reported to have a negative effect on the performance. Better results are obtained by including the labels from categorized referenced document in the indexing. This approach is reported to obtain a precision of 79%.

Larkey presents a system in [10] for searching and categorizing U.S. patent documents. The system uses a *kNN* (*K-Nearest Neighbor)* approach to categorize patents into a scheme containing around 400 classes and 135,000 subclasses. Larkey concludes that the best performance is obtained by using a vector, made up of the most frequent terms from the title, the abstract, the first twenty lines of the background summary, and the claims, with the title receiving three times as much weight as the rest of the text.

Koster et al. present in [8,9] some of the best published results on patent categorization using the *Winnow* algorithm. *Winnow* is a *mistake driven* learning algorithm that iterates over the training documents and computes for each category a vector of weights for approximating an optimal linear separator between relevant and non-relevant patents. Winnow is trained and tested with patent databases obtained from the *European Patent Office (EPO).* In the experiments with Winnow, documents are represented as a bag of words and in contrast to [10] the internal structure of the documents is completely ignored. When Winnow is tested assigning only one category per document (*mono-categorization*), it achieves a precision exceeding 98%. To achieve such high precision as much as 1000 training examples for each of the 16 categories are utilized. When the amount of training examples is reduced to 280 documents per category the precision decreases to 85%. The F_1-measure was employed for evaluating Winnow's performance when set to categorize documents that belong to more than one category (*multi-categorization*). The F_1-measure is a standard measure used in information retrieval that combines precision and recall into a single value. The optimal performance obtained on multi-categorization is an F_1-measure of 68%. This result was obtained using 88,000 training examples distributed so that each of the 44 directorates[2] has 2000 examples. It is argued that this considerable decrease in performance (larger for multi than mono-categorization) is caused by noise, since training documents are labeled arbitrarily in the border cases.

In [6] different text categorization methods included in the *Rainbow* and *SNoW package* are tested on the WIPO-alpha document collection. The Rainbow package implements *Naïve Bayes*, *kNN,* and *Support Vector Machines (SVM)* algorithms. The SNoW package implements a network of linear functions where a variation of the Winnow algorithm is used for learning. In the Rainbow package indexing is performed at word level, accounting for term frequencies in each document. The output from all classifiers consists of a ranked list of categories for each test document. In the evaluation presented in [6] three different evaluation measures are used to asses the performance of the categorization process at class-level and at subclass-level. At class level the best performance is achieved when the first 300

[2] A directorate is an administrative defined cluster.

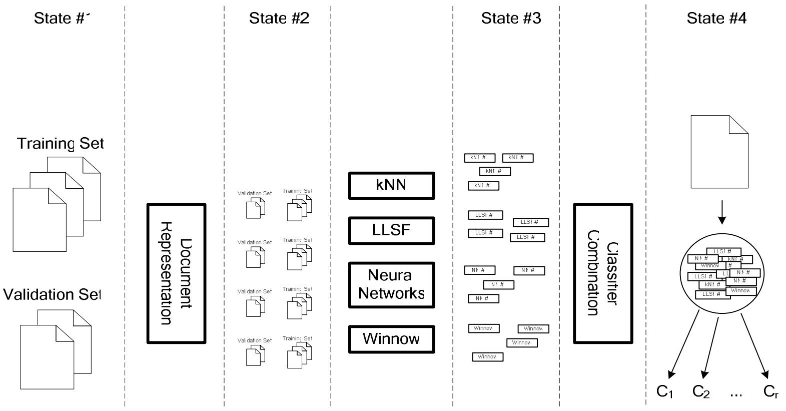

Fig. 2. The model used for constructing a combined classifier

words of each document are indexed. The best scoring text classification methods were Naïve Bayes and SVM with a precision of 55%, whereas Winnow and kNN only achieved a precision of 51%. The research in [6] revealed that the distribution of errors of Naïve Bayes is strictly different from the error distribution in SVM. Using three-guesses the best scoring method is Naïve Bayes with a precision of 79%. The precision for the other algorithms is 77% (kNN), 73% (SVM) and 73% (SNoW). When measuring with all-categories Naïve Bayes still achieves the best precision at 63%. At subclass level the best performance is also achieved when the first 300 words are indexed. Here Naïve Bayes achieves the lowest precision of all TC methods tested with a top-prediction of 33% compared to 41% best achieved precision by SVM. In first three guesses kNN achieves the best precision of 62% and tested with all-categories SVM achieves the best precision of 48%. In a second article by Fall et al. [5], a customized language independent text classification system for categorization in the IPC is presented. The system is based on state-of-the-art Neural Network techniques, and applies in particular a variant of the Winnow algorithm.

To our knowledge, no previous work has investigated automatic patent categorization methods that rely on classifier combinations. The contribution presented in this paper is to explore the effectiveness of applying diverse combination techniques trained with different document representations to categorize patents.

3 Patent Categorization Model

The combined classifier proposed in this paper is constructed in several steps. The output of each step is shown in the states depicted in Fig. 2, where the rectangular boxes represent the software components responsible for the transformation between two states. The combined classifier is able to categorize patent documents in the categories represented in the document collection (State #4 in Fig. 2). The document collection is divided into a training set, a validation set, and a test set. The training set and the validation set are used to build the classifiers (State #1 in Fig. 2) and the test set is used to evaluate classifiers' performance. The *Document Representation* component is responsible for constructing different representations of the document collections (State #2). In our experiments the document representations used vary according to three characteristics a) how features are indexed, b) how features are represented, and c) how the process of feature reduction is performed. From each representation of the document collection, four classifiers are constructed (State #3). The classifiers are trained with one of the following machine learning methods: *kNN, LLSF (Linear Least Square Fit), Neural Networks* and *Winnow*. The details of these algorithms can be found in [9,15,16].

The *Document Representation* stage consists of three separated processes: *Feature Indexing*, *Feature Weighting*, and *Feature Dimensionality Reduction*. The *Feature Indexing* process includes methods for stop-word removal and stemming and basically selects different document features to index documents. In the *Feature Weighting* process term frequencies can be used as feature weights but also other weighting schemes such as different versions of the *term frequency – inverse document frequency (tf-idf)* were included in our model. Methods for reducing the dimension of the feature collection were included in the *Feature Dimensionality Reduction* process. Our model employs two methods of *Feature Reduction*: by *Document Frequency* and by *Relevance Score*. In *Feature Reduction by Document Frequency* an upper and a lower bound obtained experimentally determines which features are included. *Feature Reduction by Relevance Score*, described in detail in [18], calculates for each feature a relevance score in each category. This score reveals how discriminative a feature is for one category in relation to all the other categories.

Our flexible patent categorization model includes also different methods to combine the machine learning based classifiers. Following sections briefly describe the methods employed to combine these classifiers; more details can be found in [16].

3.1 Combination Methods

In *Binary Voting*, voting is used to decide whether a document belongs or not to a category. Using this method it is possible to assign a document to several categories, since a voting round is conducted per category.

Weighted Classifier Combination uses the *Importance Weighted OWA* operator [14] to combine the prediction of several classifiers. The OWA operator is an averaging operator and its properties are defined by the quantifier *andness* ρ_Q applied to the algorithm using the OWA weights $\vec{w}$. Each classifier generates a value $y_{\mathrm{i}} \in [0, 1]$ signifying a document's relationship to a category c_i and for each category it also contains a value $p \in [0, 1]$ representing the precision of the classifier on the validation set. In the combination method only classifiers producing a value exceeding some threshold are averaged for each main class. Thus, the input $\vec{a}$ to the *OWA* operator is the precision obtained by the classifiers exceeding the threshold in the specific main class. Additionally, a value v_i associated with y_i is used as importance weight for the respective value a_i. The computed average is multiplied with a value b representing the number of votes k_i. The values v_i and b have associated significance scores s_y and $s_v \in [0, 1]$, which can be used to grade the impact of y_i and k_i respectively.

Dynamic Classifier Selection is based on an approach for hand-printed digit recognition proposed by Sabourin et al. [13], which selects the classifier that correctly categorizes the most consecutive neighboring training examples to perform the final prediction. In case of a tie, the algorithm implemented in our model, performs a voting round between the classifiers holding the tie, predicting the category with the highest number of votes.

Adaptive Classifier Selection was introduced by Giacinto and Roli in [7]. This method predicts also according to the best performing classifier on the validation examples in the neighborhood of the document that will be categorized. The performance of the classifier is measured according to a *soft* probability, used to identify the classifier that obtains the highest probability in categorizing a document correctly. A *confidence score*, defined as the difference in probabilities obtained by a classifier and the others, is calculated. If the confidence score exceeds a threshold, the classifier with the highest probability is used to categorize the document. If any of the computed confidence scores does not exceed the threshold, the algorithm identifies all classifiers with differences in a range of the best classifier and performs a voting round between these classifiers.

3.2 Expert Advice Algorithms

Five different *Expert Advise* algorithms were implemented in our model: *WM* [12], *WMG* [12], *P* [4], *BW* [3] *and BW'* [3]. Additionally a *mistake driven* variation of P, denoted P', was also implemented. These algorithms aim at finding the optimal combination of *experts* by minimizing the number of mistakes over a worst case sequence of observations. The idea behind the expert advice algorithms is to optimize, in a series of trials, a set of weights used to properly combine the prediction of each expert. Based on the weighted linear combination of the prediction of each expert, the algorithm is able to predict if an unseen document belongs to a category or not.

4 Experimental Results

A comprehensive collection of classifier combinations was evaluated in our experiments using different document representations. As is described in [16], 23 different document representations were tested. The 4 document representations, where the classifiers obtained the best performance, were selected for training. Fig. 3 shows the performance obtained by the classifiers on six of these document representations. The document representation that obtained the best performance (DR23), from all the categorization methods within each of the 10 main classes comprised in the refined version of the WIPO-alpha document collection, used the following features: a) *indexed sections*: title, 200 first words from abstract, 200 first words from claims and 400 first words from description, b) *feature reduction*: using 500 features per main class, c) *section weight:* the title was weighted five times as much as the other sections, and d) *feature weigh*t: normalized term frequency weight was employed. It was also determined experimentally that the best performing of the classifiers participating in the combination was *LLSF*, which achieved an F_1 measure

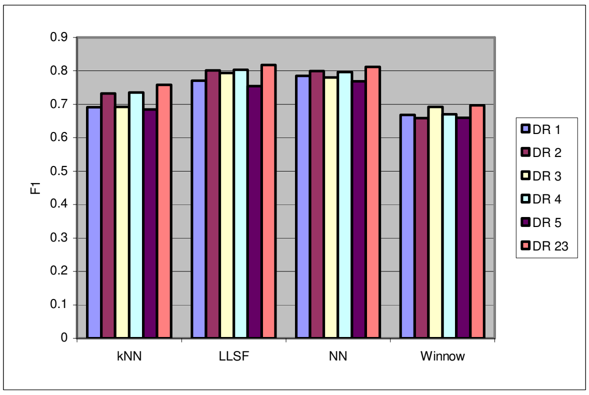

Fig. 3. Performance of four classifiers on six document representations.

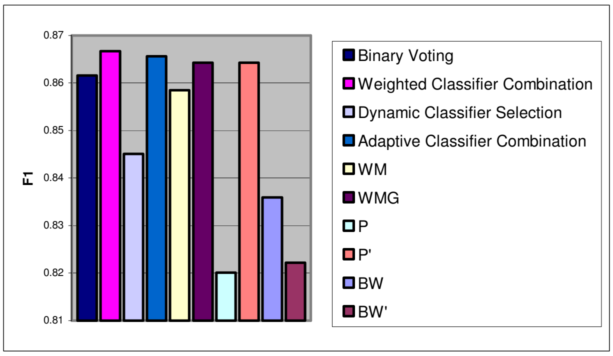

Fig. 4. F_1 measures obtained using the evaluated combination methods.

of 0.8137. The performance measures obtained by the different combination methods that were evaluated are depicted in Fig. 4. The combination method that achieved the best performance in the evaluations was *Weighted Classifier Combination* with an F_1 measure of 0.8667, which is an improvement of 6.51% compared to the best classifier participating in the combination. *Weighted Classifier Combination* achieves the best performance when the significance of voting is maximized, i.e. the voting phase of the algorithm is favored. Similar properties were observed with other combination methods. The great impact of voting in performance might be caused by the relatively large number of classifiers participating in the combinations.

Our results also show that the performance obtained by classifiers based on Winnow and kNN is inferior to the performance obtained by classifiers based on LLSF and Neural Networks. To determine the impact on performance of these classifiers, two of the combination methods were evaluated with and without classifiers based on Winnow and kNN participating in the combination. The evaluated combinations showed more effectiveness without a classifier based on kNN but including classifiers based on Winnow. To evaluate the contribution of the four

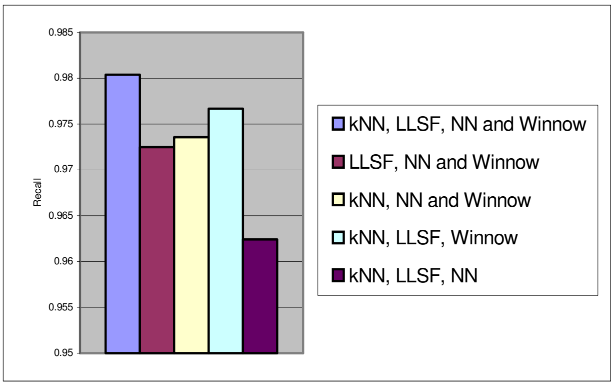

Fig. 5. Recall obtained using *BestSelect* with different combinations of categorization

categorization methods the *BestSelect* algorithm described in [1] was applied on five collections of classifiers. *BestSelect* predicts correctly whenever any of the classifiers in the combination predicts also correctly; otherwise the prediction remains undefined. We applied *BestSelect* on five settings, each comprising the same document representations where either, all of the categorization methods or all methods except the method from which the contribution should be tested, participated in the combination. The difference between the recall measured on a setting where classifiers based on one method are not contained in the combination and the setting where all classifiers are contained in the combination, can be seen as the contribution of a particular method. The recall obtained using *BestSelect* on the five settings is depicted in Fig. 5. This figure surprisingly reveals that combinations including a *Winnow* classifier have the best performance, although this classifier alone showed relatively poor performance.

5 Conclusions and Future Work

In this paper we have described a new model of an automatic patent categorization system based on an ensemble of classifiers. Our model was evaluated on a refined version of the WIPO-alpha document collection. However, since no previous research has been evaluated on this same collection no direct comparison with other methods can be done at this time. Instead, the performance results obtained by the ensemble were compared to the best performing categorization method used in the combination. As is described in Section 3, we evaluated 10 different techniques for combining the classifiers. Our experiments show that all of the combination methods achieved improved performance when compared to the best classifier participating in the combination. The best combination technique *Weighted Classifier Combination* achieves an F_1 score of 0.8667, which is an improvement of 6.51% to the best classifier participating in the ensemble. Among the four machine learning based categorization methods, the best performing were *Neural Networks* and *LLSF*, but they also have the worst training efficiency. However, their training can be improved reducing the feature collection. Interestingly, our experiments also show that reducing the feature collection improves the performance of some classifiers. The categorization methods employed in our model are representative of a variety of the available methods. As future research we plan to explore combining a broader class of classifiers and optimize the overall performance of the ensemble using genetic algorithms.

References

1. Bennett P.N., Dumais S. T., and Horvitz E.: Probabilistic Combination of Text Classifiers Using Reliability Indicators: Models and Result. Proceedings of the *25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'02)*, Tampere, Finland, 2002.
2. Chakrabarti S., Dom B., and Indyk P.: Enhanced hypertext categorization using hyperlinks, Proceedings of SIGMOD98, *ACM International conference on Management of Data*, ACM Press, New York, 307-318, 1998.
3. Cesa-Bianchi N., Freund Y., Helmbold D. P., and Warmuth M. K.: On-line Prediction and Conversion Strategies, *Machine Learning* 25, pp. 71-110, 1996.
4. Cesa-Bianchi N., Freund Y., Hausler D., Helmbold D. P., Schapire R E. , and Warmuth M. K.: How to Use Expert Advice, *Journal of the ACM*, Vol. 44, No. 3, May 1997, pp. 427-485.
5. Fall C. J., Benzineb K., Goyot J., Törcsvári A., and Fiévet P.: Computer-Assisted Categorization of Patent Documents in the International Patent Categorization, *Proceedings of the International Chemical Information Conference*, Nîmes, October 2003.
6. Fall C. J., Törcsvári A., Benzineb K., and Karetka G.: Automated Categorization in the International Patent Classification, *ACM SIGIR Forum*, Vol 37(1), 10-25, 2003.
7. Giacinto G. and Roli F.: Adaptive Selection of Image Classifiers, *In Proceedings of ICIAP*, *Springer Verlag LNCS,* Vol. 1310, pp 38-45, 1997.
8. Koster C.H.A., Seutter M., and Beney J.: Classifying Patent Applications with Winnow, Proceedings Annual Machine Learning Conference Benelearn, Univ. Antwerp 2001.
9. Koster C.H.A., Seutter M. and Beney J.:Multi-categorization of Patent Applications with Winnow, *Ershov Memorial Conference 2003*, Novosibirsk, Russia, 546-555, 2003.
10. Larkey L. S.: A Patent Search and Categorization System, *Proceedings of the 4th ACM Conference on Digital Libraries*, 179-187, 1999.
11. Larkey L. S. and Croft W. B.: Combining Classifiers in Text Categorization, *In Proceedings of SIGIR-96, 19th ACM conference on Research and Development in Information Retrieval,* Zürich, Switcherland, 1996, p. 289-297.
12. Litlestone N. and Warmuth M. K.: The Weighted Majority Algorithm, *Information and Computation,* Vol. 108, 211-261, 1994.
13. Sabourin M., Mitiche A., Thomas D., and Nagy G.: Classifier Combination for Hand-printed Digit Recognition, *Proc. Second Int. Conf. Document Analysis and Recognition*, pp. 163–166, Tsukuba Saenie City, Japan, 20–22 Oct. 1993.
14. Yager R.R.: On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Transactions on Systems, Man and Cybernetics* 18(1988) 183-190.
15. Yang Y.: An Evaluation of Statistical Approaches to Text Categorization, *Information Retrieval,* Vol. 1, 69-90, 1999.
16. Mattiason H.: Automated categorization of Patent Applications, MSc Thesis. Computer Science and Engineering Department, Aalborg University Esbjerg, June 2006.