



Automated Single-Label Patent Classification using Ensemble Classifiers

Eleni Kamateri

Department of Information and Electronic Engineering,
International Hellenic University (IHU), Greece
ekamater@hotmail.com

Konstantinos Diamantaras

Department of Information and Electronic Engineering,
International Hellenic University (IHU), Greece
k.diamantaras@ihu.edu.gr

Vasileios Stamatis

Department of Information and Electronic Engineering,
International Hellenic University (IHU), Greece
vstamatis@it.teithe.gr

Michail Salampasis

Department of Information and Electronic Engineering,
International Hellenic University (IHU), Greece
msa@ihu.gr

ABSTRACT

Many thousands of patent applications arrive at patent offices around the world every day. One important task when a patent application is submitted is to assign one or more classification codes from the complex and hierarchical patent classification schemes that will enable routing of the patent application to a patent examiner who is knowledgeable about the specific technical field. This task is typically undertaken by patent professionals, however due to the large number of applications and the potential complexity of an invention, they are usually overwhelmed. Therefore, there is a need for this code assignment manual task to be supported or even fully automated by classification systems that will classify patent applications, hopefully with an accuracy close to patent professionals. Like in many other text analysis problems, in the last years, this intellectually demanding task has been studied using word embeddings and deep learning techniques. In this paper these research efforts are shortly reviewed and re-produced with similar deep learning techniques using different feature representations on automatic patent classification in the level of sub-classes. On top of that, an innovative method of ensemble classifiers trained with different parts of the patent document is proposed. To the best of our knowledge, this is the first time that an ensemble method was proposed for the patent classification problem. Our first results are quite promising showing that an ensemble architecture of classifiers significantly outperforms current state-of-the-art techniques using the same classifiers as standalone solutions.

CCS CONCEPTS

• **Social and professional topics** -> **Computing / technology policy** -> **Intellectual property** -> **Patents**; • **Computer systems organization** -> **Architectures** -> **Other architectures** -> **Neural networks**; • **Computing methodologies** -> **Machine learning** -> **Learning paradigms** -> **Supervised learning** -> **Supervised learning by classification**;



This work is licensed under a Creative Commons Attribution International 4.0 License.

ICMLC 2022, February 18–21, 2022, Guangzhou, China

© 2022 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-9570-0/22/02.

<https://doi.org/10.1145/3529836.3529849>

KEYWORDS

Patent, Classification, Single-label, Sub-classes, Ensemble method, Deep learning, Word embeddings

ACM Reference Format:

Eleni Kamateri, Vasileios Stamatis, Konstantinos Diamantaras, and Michail Salampasis. 2022. Automated Single-Label Patent Classification using Ensemble Classifiers. In *2022 14th International Conference on Machine Learning and Computing (ICMLC) (ICMLC 2022), February 18–21, 2022, Guangzhou, China*. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3529836.3529849>

1 INTRODUCTION

Patents have increasingly become an important economic asset and patent filing rates have enormously increased in recent years. In 2019, innovators filled about 3.2 million patent applications worldwide [1]. Patent offices internationally must deal with these large number of applications. Therefore, automating any subtask of the large patent examination process is an important challenge which has significant impact because it can speed up the examination process. One subtask that can be automated is the pre-classification which is typically done by front-line patent experts who manually classify an application using a classification scheme (e.g., IPC, CPC). This task is quite important considering that this classification will route the patent application to a sub-department of the office for detailed examination. Also, during the examination process, properly assigning other relevant classification codes ensures that patents with similar technical features will be grouped under the same intellectual scheme, something which is crucially important for many subsequent patent retrieval tasks. For example, in prior art search [2], the assigned codes could substantially improve the search for relevant patents in different ways such as filtering the search or expanding with extra search terms.

Classification schemes follow a hierarchical structure meaning that each inner node in the hierarchy has exactly one parent and the path to each code is unique [3]. The most widely used classification scheme is the International Patent Classification (IPC) which follows this hierarchical structure containing thousands of codes each representing a more general (at higher levels) or very specific technological concept. In the version of IPC 2006 for example, the classification scheme contains in total 8 sections, 131 classes, 642 sub-classes, 7,537 groups and 69,487 subgroups.

Two important features for automated classification are the following: a) it can be done at different levels of the classification hierarchy and b) it can be done either as a single-label task in case of the pre-classification stage (described above) or as a multi-label task because each patent can have multiple codes assigned, sometimes at a different level of the hierarchy. Several research works tried to automate the patent classification task, however because automated systems do not attain human performance, the automation only worked at a higher level of the hierarchy.

Recently, automated patent classification, which falls under the broad field of text classification [4, 5], has been examined using deep learning (DL) methods such as convolutional neural networks (CNN) [6–8], Word2Vec and Long-short term memory (LSTM) [9–11] and other DL methods [12] to predict the most representative classification code(s) for a patent. In the current work, these CNN and RNN models are evaluated employing different language models and different parts of the patent document. Additionally, an ensemble method is proposed combining the performance of standalone CNN and RNN classifiers trained at different parts of the patent document. This study focuses on single-label classification at the sub-class (3rd) level category of the IPC 5+ level hierarchy. The sub-class level was selected because this is the level in which the pre-classification task is typically performed. Moreover, the sub-class level is the minimum level in the IPC hierarchy where it has value to use an automated system, although the group (4th) and sub-group (5th) level are even more useful. However, it should be noted that classifying patents at the sub-class is already a very difficult task, because at this level there are approximately 650 labels, much more than those found in a typical text classification problem. Furthermore, the classification task becomes even more complex considering that only one main label should be accurately assigned to each patent.

To create a proper dataset for evaluating the proposed methods for single-label classification, the <main classification> tag is used that exists in patents in the CLEF-IP 2011 dataset to extract the primary classification. This label denotes the main and most important classification code assigned to a patent. Keeping only the patents that have a main classification tag assigned, a subset of the CLEF-IP collection is created.

The remainder of the paper is structured as follows: Section 2 describes related work in the area of automatic patent classification. Section 3 describes the proposed ensemble method and the individual CNN and RNN classifiers taking place in the ensemble architecture. Section 4 presents the experimental methodology and setup to evaluate the proposed method. Section 5 presents and discusses the experimental results. Finally, section 6 concludes the paper and discuss the ongoing and future work.

2 PRIOR WORK

Several works were presented to solve the automated patent classification problem. Actually, patent classification methods advance hand-to-hand with the text and document classification that are typical Natural Language Processing (NLP) applications which deal with the assignment of one or multiple pre-defined labels to a given text or document respectively [8].

Earlier research on patent classification applied basic NLP and feature engineering techniques to pre-process texts before feeding them into well-known classifiers. For example, Fall et al. [13] performed stop words removal, stemming, term selection using the information gain and then fed the transformed texts into Naïve Bayes (NB), Support Vector Machine (SVM), and K-Nearest Neighbor (KNN) classifiers. Tikk et al. [14] applied stop word removal, stemming, dimensionality reduction and removal of rare terms, and they sent the processed output into a neural network, named HITEC, which copies the tree-like structure of the taxonomy. Their method showed 53.25% classification accuracy at the sub-class level.

Around 2017, research on automated patent classification turned to test out the effectiveness of new DL methods for text processing tasks. Among them, Grawe et al. [9] used stop word removal and then transformed the cleaned text into a meaningful representation produced by the Word2Vec that was sent as input to an LSTM network (a type of RNN). The method achieved 63% accuracy at the sub-class level. Likewise, Xiao et al. [10] used Word2Vec embeddings and LSTM to classify patents in the security field.

Another recent trend in the relevant literature is the training of Word Embeddings on domain-specific patent datasets. Risch and Krestel [12] used the FastText embedding that was trained on a patent dataset together with bi-directional GRUs (another type of RNN) to achieve better performance compared with Word Embeddings trained on Wiki documents. Their method achieved 53% accuracy at the sub-class level. Moreover, Sofean [11] developed a self-trained Word Embedding trained on million patents and then used a LSTM network to perform patent classification. For evaluating his method, Sofean kept only the sub-classes appearing in more than 500 documents, resulting to 43 sub-classes, and obtained an accuracy of 67%.

Another improvement in patent classification with respect to the DL techniques started with the adoption of CNNs. Li et al. [6] proposed a DL algorithm, DeepPatent, for patent classification by combining word vector representation and a well-designed CNN. DeepPatent performs multi-label classification at the sub-class level getting 75.46% recall at top 4. Likewise, Zhu et al. [7] used the Word Embedding technique to segment and vectorize the input data and then a symmetric hierarchical CNN, named PAC-HCNN, to classify patents outperforming traditional RNN. Moreover, Abdelgawad et al. [8] compared several recent neural network models and showed that CNNs are a suitable choice for patent classification. The authors also proved that state-of-the-art hyperparameter optimization techniques can further improve the CNN performance getting an accuracy of 52.02 at the sub-class level.

Recent trends on DL that includes pre-trained unsupervised language models on large corpuses and fine-tuning them on downstream tasks have produced state-of-the-art performances. Among them, the BERT model, released by Google in 2018, is the first deeply bidirectional, unsupervised language representation, pre-trained using only a plain text corpus in order to infuse later the context in which it will be used. Following this trend, Lee et al. [15] leveraged and fine-tuned the BERT-Base model and applied it to patent classification getting better accuracy than other recent DL approaches. This method was evaluated on a multi-label classification and achieved 54.33 recall at 1. Similarly, Roudsari et al.

[16] presented a short work applying and fine-tuning Distil-BERT model for patent classification.

Last but not least, some researchers tried to identify which parts of a patent document can provide more representative information for classification tasks [15, 17]. They showed that the use of technical and background parts extracted from the descriptions section [17], the first claim [15], the title and abstract [6, 7], or at most cases, the title, abstract claims and description [8, 11, 12] are the most useful parts for patent classification.

3 METHODS

An ensemble method receives evidence from multiple models, working either at the same or different sources of information, combines these evidences and produces a final prediction. This technique obtains better predictive performance than that could be obtained from any of the constituent models [18] since it exploits potentially not related information coming from all single models. Although ensemble methods are experiencing good results in many applications, they are less explored for automated patent classification and only for upper levels of the IPC hierarchy [19].

In this section, an ensemble architecture comprising of three individual neural network classifiers is presented to address the automatic patent classification problem at the sub-class level. Although the individual classifiers (i.e., CNN and RNN) can further be improved by tailoring their structure with additional layers and fine-tuning the different parameters, a simple version of these networks has been selected to be deployed because the accuracy rates are not significantly improved when more complex architectures were tested.

3.1 Ensemble architecture

The ensemble architecture consists of three individual classifiers with each of them trained on a different part of the patent text, i.e., the title-abstract, the description and the claims section, respectively. These individual classifiers can be of any type. In this study, identical classifiers have been used. Each classifier produces a list of probabilities for all labels based on its partial knowledge about the patent. Then, the probabilities for a specific label derived from the three individual classifiers are averaged and a final probability is calculated for this label. The label with the maximum probability consists the predicted label for the patent.

3.2 Individual classifiers

In this study, state-of-the-art CNN and RNN classifiers are used as individual classifiers of the ensemble architecture since, based on the literature, they are experiencing good results in automated patent classification.

3.2.1 CNN. CNN is one of the neural networks that have been widely used recently in the domain of automated patent classification. In our method, the pre-processed text is fed to a CNN through an embedding lookup, which converts words to vectors represented in a high-dimensional vector space. Afterwards, a 1-D convolutional layer is applied on top of the embedding layer. A max-pooling along with a flatten layer are then applied sequentially to the output of the convolutional layer. After a dense of 1024 filters and a dropout layer with a dropout rate of 0.5, the output is fed in

another dense layer with a softmax activation in order to obtain a probability distribution over all targeted labels (IPC sub-classes which are equal to 659).

3.2.2 RNN. RNN is another category of neural networks that has been widely used in text classification. Both LSTM and GRU have been proposed as variations of RNN to deal with the exploding gradient and vanishing gradient problems of back propagation through time experienced by RNN models. Since they consistently produce good results in text analysis tasks, they have been applied in many fields including automated patent classification. Here, both LSTM and GRU models are evaluated as they are superior to simple recurrent units. Moreover, bi-directional LSTM and GRU are used so that the input sequence to be processed in two directions of the sentence. Similar with CNN architecture, each word is converted to its embedding. This embedding layer of words is processed by a spatial dropout, which randomly masks 10% of the input words to make the neural network more robust. The remaining 90% serves as input to the LSTM or GRU layer (also to the bidirectional LSTM and bidirectional GRU). The output of these steps is inserted in a dense layer with as many units as the targeted labels and a softmax activation.

4 EXPERIMENTS

This section describes the data collection, the selected data pre-processing, the experimental methodology, the different sets of experiments conducted to address the automatic patent classification problem at the sub-class level, and the metrics for evaluating the experiments.

4.1 Data collection

For evaluating our techniques, the CLEF-IP 2011 test collection was used. From all patent documents in the test collection, only 418,788 have English texts and the required information, which is their main classification category, i.e. include the <main classification> tag. This information was mandatory to determine which is the primary classification code, so a single-label classification method can be reliably executed and evaluated. From the initial 418,788 only 302,578 patent documents have been remained as only these contain a title-abstract, a description and a claims section in the patent document at the same time.

From those patents, four different patent pools are created: i) the initial pool which contains the content from the title and the abstract section; ii) a second pool with content from the description section; iii) a third pool with content from the claims section and; iv) a last pool with content from all aforementioned sections.

4.2 Data pre-processing

Afterwards, the first "X" words from the concatenated result of the title-abstract, the description, and the claims sections are extracted. This means that most of the words used to represent the patent document are taken from the title-abstract. This leaves out a (often small) number of words which comes from the description section, while words from the claims section will be not used in the final "X" words length representation of the patent document. The same methodology is repeated in rest three "mono-thematic" data collections retrieving the first "X" words coming from title and abstract,

the description and the claims sections for each patent, respectively. For all data collections, the different number of words, called "X", that has been tested was: {20, 40, 60, 80, 100, 200, 300, 400}.

The method of retrieving the first "X" words for the representation of a patent document is followed in several literature works [6, 9, 10, 12]. Although this method was followed in the current experimental setup, we have introduced some improvements, such as the usage of a varying number of words from each patent part. Another improvement is the usage of a varying number of words from all patent parts.

4.3 Experimental methodology

For each patent, the patent target variable (i.e., sub-class IPC code) was encoded using one-hot encoding. On the other hand, the patent text, formed by the selected first "X" or "Y" words, was followed a sequence of processing steps. It was tokenized and converted into a sequence of tokens, while padding was used to ensure equal length vectors. Tokens were then mapped to embeddings and an embedding matrix representing each patent document has been created. For the transformation of tokens to embeddings, the pre-trained language models of FastText, Word2Vec, and Glove are used with dimension size set to 300 for the generated word vector. Word2Vec was trained using the genism toolkit in our corpus (the corpus of 302,712 patent documents) and then the produced embeddings were used to represent the patent texts. For the training of Word2Vec, the vector size was set to 300, the maximum distance between current and predicted word to 8 and the number of iterations to 20. The outcome embedding matrix was later used as input to the neural network architecture, which consists of several layers and returns as outcome a probability for each target.

Moreover, the dataset of 302,712 patents was split into training, validation and testing sets; 80% for training, 10% for validation, while other 10% was kept out for testing the classification model. After some tests, batch size was set to 128 and epochs to 5 for CNN models and 15 for RNN models.

4.4 Experimental setup

This experimental layout explores the effect of different deep learning models, language models and feature selection methods on patent classification results. More specifically, the first experiment evaluated how different batches of words coming from different parts of the patent document may result into different outcomes on patent classification (Exp#1). Then, the second experiment tested how different embedding representations can affect the patent classification outcomes (Exp#2). The following experiment explored the effect of different deep learning models on patent classification results (Exp#3). Last, the last experiment evaluated the ensemble method of combining the results of similar classifiers trained in parallel in the three different parts of the patent document (Exp#4).

4.5 Evaluation criteria for single-label patent classification

For each experiment, *Accuracy* and *Recall at n* are used to evaluate our methods. The selection of the specific metrics was made after a thorough review of the literature which showed that these are the only metrics that accurately resembles the actual real task of

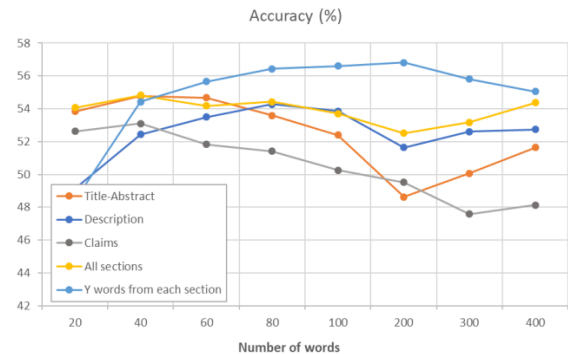


Figure 1: Accuracy scores achieved for a varying number of words retrieved from each of the four data collections (first four lines) or retrieved in parallel from all patent parts (light blue line labelled "Y words from each section").

single-label patent classification. With *Accuracy* metric, the first prediction of the classifier is evaluated for each patent document while for *Recall at n* metric, the top-n predictions for each patent document are evaluated.

5 RESULTS

5.1 Exploring different feature selection techniques for patent representation

In this set of experiments (Exp#1), the first "X" words from each patent part and the first "Y" words from all patent parts were extracted for the representation of the patent document. These words consist the input nodes of the DL model. The architecture of the CNN individual classifier, described in section 3.2.1, was used for the classification task and the FastText pre-trained word embedding was used for the text representation. Figure 1 illustrates the accuracy scores for varying numbers of retrieved words from a specific patent part each time, compared to the accuracy achieved when equal words are retrieved from each patent part simultaneously.

The accuracy is better when retrieving "Y words from each section" at the same time to represent the patent document. After, it comes the accuracy when retrieving words from "all sections", then follows the accuracy of the "title-abstract", the "description", and last comes the accuracy of the "claims". Therefore, the best representation is achieved when using words from all sections. The title-abstract and description sections are constantly important descriptors of the patent document. The claims sections have the lowest accuracy score meaning that these have low representation value in the patent document which is reasonable considering that this section defines the legal boundaries of an invention using difficult and vague vocabulary.

The scores for "all sections" and "title-abstract" section are initially quite similar, which is reasonable considering that the number of words is low and most words are retrieved from the title-abstract section. For bigger number of words, the accuracy of "all sections" become almost equal with the accuracy of the "description", which can be interpreted that the first words used in the description section are quite important for the patent representation.

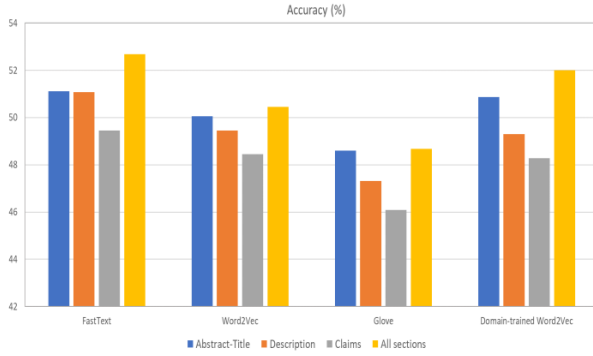


Figure 2: Accuracy scores using different language models for representing the patent document.

5.2 Exploring different language models

In this second experimental setup (Exp#2), different word embeddings were explored to evaluate how they can affect the patent classification outcome. The first 60 words from each patent part were retrieved for representing the patent documents that used as input for the DL model. Similar to Exp#1, the architecture of the CNN individual classifier (section 3.2.1) was used for the classification. In Figure 2, the accuracy scores for different language models and data collections are displayed.

Among pre-trained language models, the FastText seems to achieve better representation of the patent text and thus better accuracy scores than other two pre-trained language models (i.e., Word2Vec and Glove). For the FastText, the accuracy reaches 52.69% for the dataset of all sections. It is also interesting that even when the Word2Vec language model has been trained on the domain-specific corpus and then used for the representation of the patent text, it reaches worse results (52.01%) than FastText.

5.3 Exploring different DL methods

In this experiment the effect of different DL models on patent classification results was explored (Exp#3). As mentioned in section 3.2, the DL algorithms that were selected to be evaluated as individual classifiers in this study are a CNN, a Bidirectional LSTM, a Bidirectional GRU, a LSTM and a GRU. In this experiment, the first 60 words (input nodes) were retrieved for all data collections and the FastText pre-trained word embedding was used for the text representation. Figure 3 shows the accuracy scores achieved for different DL models for all data collections.

It is shown that LSTM and GRU models perform similar, while a significant improvement in accuracy is achieved by the usage of the bidirectional DL models (either GRU and LSTM), which demonstrate the best accuracy scores among all data collections. More specifically, the bidirectional LSTM reaches the best accuracy score of 59.83% for the data collection of the title and abstract. The bi-directional LSTM achieves better classification performance than almost all similar state-of-the-art methods [8, 12, 15], except for [9] which exploits a language model trained on a domain-specific dataset.

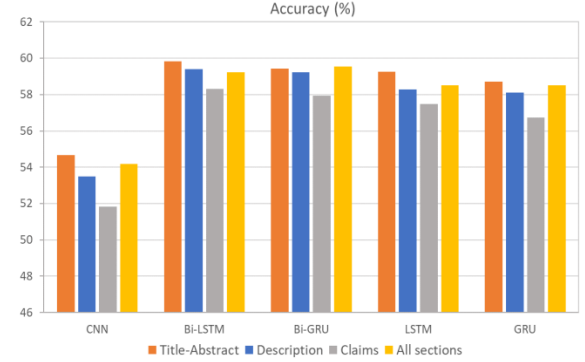


Figure 3: Accuracy scores using different DL models for patent classification.

5.4 Ensemble method

In this experimental setup (Exp#4), a simple ensemble method averaging the outcome of three individual classifiers, each applied on a “mono-thematic” data collection, was explored to evaluate whether it can achieve better classification results than each of those classifiers acting on its own.

Table 1 presents the evaluation metrics of each individual classifier trained on the 60 words coming from the title-abstract, the description and the claims part, respectively, and the evaluation scores of the ensemble method combining the results of the above three individual classifiers. In order to quantify the improvement experienced with ensemble method, Table 1 also presents the average accuracy achieved by three individual classifiers compared with the accuracy achieved by the ensemble method of them.

Table 1 shows that evaluation scores are much improved when an ensemble method is applied compared with scores archived by individual classifiers working on each patent part. More specifically, the ensemble of bidirectional GRU classifiers achieved an accuracy of 65%, which is the best accuracy score compared with the scores achieved by individual classifiers. Moreover, the ensemble architecture of bidirectional GRU classifiers provides the best accuracy score compared with the other combinations of neural network classifiers.

Figure 4 and Figure 5 show the recall at first three and at first five returned results, respectively, for individual classifiers and the combination (ensemble) of them.

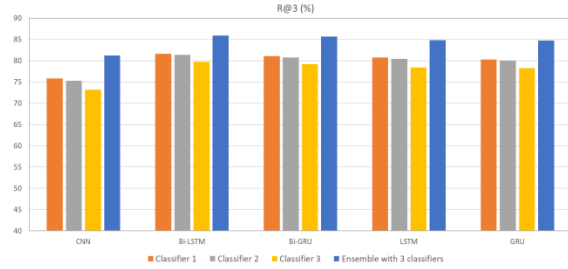
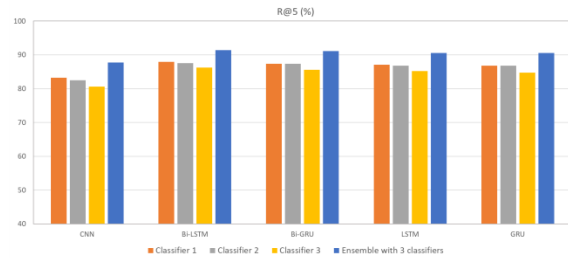
The most interesting observation in these figures is that the recall at n increases significantly when the top three and five returned targets are evaluated. Specifically, it reaches 85.88% and 91.35%, respectively, when bi-directional LSTM was used as individual classifiers and trained on different patent parts. Moreover, the ensemble method of three identical classifiers trained with different patent text achieves better classification accuracy compared with the results provided by state-of-the-art research efforts [8, 9, 12, 15].

6 CONCLUSION

This work clearly illustrates that an ensemble method of classifiers using the optimal DL models and domain parameters produces better results compared with standalone classifiers. More specifically,

Table 1: Accuracy scores achieved by individual classifiers and the ensemble of them

	CNN	Bi-LSTM	Bi-GRU	LSTM	GRU
Accuracy of individual classifier 1 applied on title-abstract	53.65	59.83	59.43	59.26	58.71
Accuracy of individual classifier 2 applied on description	52.99	59.40	59.24	58.28	58.10
Accuracy of individual classifier 3 applied on claims	51.54	58.31	57.93	57.48	56.72
Average accuracy of individual classifiers applied 1, 2 and 3	52.73	59.18	58.87	58.34	57.84
Ensemble method combining individual classifiers 1, 2 and 3	59.54	64.57	64.85	63.51	63.44

**Figure 4: R@3 score using individual classifiers and an ensemble method.****Figure 5: R@5 score using individual classifiers and an ensemble method.**

an ensemble architecture of three bidirectional GRU classifiers, working on different patent parts simultaneously and combining their results, achieved the highest accuracy score of 65%, which is much improved compared with the accuracy achieved when each of these classifiers working on its own. Moreover, the ensemble of individual classifiers managed to predict the correct label among the top three, five and ten returned labels with an accuracy of 85.88%, 91.35 and 95.65%.

Furthermore, the evaluation of different DL models showed that a bidirectional-LSTM or a bidirectional-GRU can achieve better results than other DL methods, especially when it is combined with FastText word embeddings. With respect to the patent representation, patent sections from which the feature words are retrieved and the number of words selected to represent the patent document also seem to play an important role for automated patent classification. Although the abstract section offers the most valuable words for automated patent classification, the description section contains also important information that makes a patent distinguishable. Moreover, the selection of feature words from all parts of a patent

at the same time achieves better results than the selection of the first words from each part.

These results are quite promising for producing an effective patent classification tool that can support patent officers in the pre-classification task suggesting a list of potential IPC codes. Our future plan is to continue the work on ensemble methods focusing on different strategies for combining different individual classifiers' results.

ACKNOWLEDGMENTS

This work has received funding from the EU Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No: 860721 (DoSSIER Project, <https://dossier-project.eu/>).

REFERENCES

- [1] WIPO, World intellectual property indicators 2020, 2021. Available at: https://www.wipo.int/edocs/pubdocs/en/wipo_pub_941_2020.pdf.
- [2] Salampasis, M., Paltoglou, G., Giahannou A. (2012). Report on the CLEF-IP 2012 Experiments: Search of Topically Organized Patents. CLEF (Online Working Notes/Labs/Workshop)
- [3] B. Wolter. It takes all kinds to make a world—some thoughts on the use of classification in patent searching. *WPI*, 34(1), pp. 8-18, 2012
- [4] Nguyen, T. T., Hatua, A., & Sung, A. H. (2017). Building a learning machine classifier with inadequate data for crime prediction. *Journal of Advances in Information Technology Vol*, 8(2).
- [5] Nam, J. H., & Yow, K. C. (2017). A Data-driven Approach to the Automatic Classification of Korean Poetry. *Journal of Advances in Information Technology Vol*, 8(3).
- [6] Li, S., Hu, J., Cui, Y., & Hu, J. (2018). DeepPatent: patent classification with convolutional neural networks and word embedding. *Scientometrics*, 117(2), 721-744.
- [7] Zhu, H., He, C., Fang, Y., Ge, B., Xing, M., & Xiao, W. (2020). Patent Automatic Classification Based on Symmetric Hierarchical Convolution Neural Network. *Symmetry*, 12(2), 186.
- [8] Abdelgawad, L., Kluegl, P., Genc, E., Falkner, S., & Hutter, F. (2019). Optimizing neural networks for patent classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 688-703). Springer, Cham.
- [9] Grawe, M. F., Martins, C. A., & Bonfante, A. G. (2017). Automated patent classification using word embedding. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)* (pp. 408-411). IEEE.
- [10] Xiao, L., Wang, G., & Zuo, Y. (2018). Research on patent text classification based on word2vec and LSTM. In *2018 11th International Symposium on Computational Intelligence and Design (ISCID)* (Vol. 1, pp. 71-74). IEEE.
- [11] Mustafa Sofean (2019) Deep Learning based Pipeline with Multichannel Inputs for Patent Classification. 1st Workshop on Patent Text Mining and Semantic Technologies. PatentSemTech
- [12] Risch J. & Krestel, R. (2019). Domain-specific word embeddings for patent classification. *Data Technologies and Applications*.
- [13] Fall, C. J., Töröcsvári, A., Benzineb, K., & Karetka, G. (2003). Automated categorization in the international patent classification. In *Acm Sigir Forum* (Vol. 37, No. 1, pp. 10-25). New York, NY, USA: ACM.
- [14] Tikk, D., Biró, G., & Töröcsvári, A. (2008). A hierarchical online classifier for patent categorization. In *Emerging technologies of text mining: Techniques and applications* (pp. 244-267). IGI Global.

- [15] Lee, J. S., & Hsiang, J. (2020). Patent classification by fine-tuning BERT language model. *World Patent Information*, 61, 101965.
- [16] Roudsari, A. H., Afshar, J., Lee, C. C., & Lee, W. (2020). Multi-label patent classification using attention-aware deep learning model. In *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)* (pp. 558-559). IEEE.
- [17] Lim, S., & Kwon, Y. (2016). IPC Multi-label Classification Based on the Field Functionality of Patent Documents. In *International Conference on Advanced Data Mining and Applications* (pp. 677-691). Springer, Cham.
- [18] Zhou, Z. H., Wu, J., & Tang, W. (2002). Ensembling neural networks: many could be better than all. *Artificial intelligence*, 137, 239-263.
- [19] Benites, F., Malmasi, S., & Zampieri, M. (2018). Classifying patent applications with ensemble methods. *arXiv preprint arXiv:1811.04695*.