Japanese Patent Classification Using Few-shot Learning

Shota Hachisuka
Graduate School of Science and Engineering
Chuo University
Tokyo, Japan
a19.j83d@g.chuo-u.ac.jp

Yuta Nakada
Graduate School of Science and Engineering
Chuo University
Tokyo, Japan

Hidetsugu Nanba
Faculty of Science and Engineering
Chuo University
Tokyo, Japan
nanba@kc.chuo-u.ac.jp

Satoshi Fukuda
Faculty of Science and Engineering
Chuo University
Tokyo, Japan
fukuda.satoshi.3238@kc.chuo-u.ac.jp

***Abstract*—In general, complex search formulae are manually created in patent search by combining keywords with classification codes such as F-term, and the target patents are retrieved using these formulae. Then, the obtained results are manually checked one by one to collect the target patents. Because the manual checking process is time-consuming, an automatic classification method is required. Recently, deep learning has been widely used for document classification. However, this requires a large amount of training data, which is not available due to the cost of data preparation. We address this problem by using few-shot learning to construct a classifier that can efficiently narrow down the target patents. The experimental results confirm the effectiveness of the proposed method.**

***Keywords*—few-shot learning, patent classification, patent, text classification, F-term**

I. Introduction

In this study, we construct a patent classifier to efficiently retrieve patents in a specific domain by using few-shot learning, which can be trained with a small amount of training data [1]. In general, complex search formulae are manually created in patent search by combining keywords with patent classification codes, and the target patents are retrieved using these formulae. Then, the obtained results are manually checked one by one to collect the target patents. When a company develops a product over a long period, this series of search operations must be repeated during the development period. Because it is very costly to manually check the search results, there is a need to automate this checking process.

The task of looking for target patents among search results refers to document classification. In recent years, deep learning has been widely used for document classification. However, this technique requires a large amount of training data, which is often not available due to the cost of data preparation. To address this problem, we construct a patent classifier based on few-shot learning, which is capable of learning even with a small amount of training data.

Few-shot learning is divided into two stages: pretraining and fine-tuning. If a large amount of training data is available for pretraining, a small amount of training data is sufficient for fine-tuning if the task of pretraining is close to that of fine-tuning. Therefore, we construct a patent classifier, which identifies whether each patent retrieved is a patent that the searcher is looking for. We do this using a small amount of training data by conducting pretraining with a large number of patents with F-term, which is a category for classifying Japanese patents.

II. Related Work

A. Few-shot Learning

Few-shot learning is a machine learning method that transfers knowledge learned with abundant training data in one domain to training another domain. Few-shot learning is costly because it requires training on a large set of documents related to the specific task in advance, but it can be an effective method for data with very small samples. In the field of computer vision, Snell et al. [2] proposed distance learning, which can classify images by calculating the distance between the prototype representation of each class and the query in few-shot learning. Few-shot learning includes a method called meta-learning, which aims to learn a learning method from data related to the target task and improve the performance of the target task. Chen et al. [3] experimentally confirmed that fine-tuning-based few-shot learning outperforms a meta-learning-based approach [4]. Therefore, we also employ fine-tuning-based few-shot learning in patent classification.

B. Text Classification

Li et al. [5] proposed a patent classification for F-term using support vector machine (SVM). They extracted nouns, verbs, adjectives, and unknown words from patents and used them as features for the SVM. We employ few-shot learning, and compare the result with that of SVM.

Task-dependent problems often occur in few-shot learning, because lexical features that are useful in one task are not always useful in other tasks. To address this problem, Yu et al. [6] improved few-shot learning by taking account of the similarities between tasks in few-shot learning. In our work, we assume that the task in the pretraining stage is similar to that in the fine-tuning stage.

GPT-3 is a pretrained natural language processing model that enables few-shot learning [7]. To apply GPT-3 to our study, it is necessary to input the few-shot data; however, there is an upper limit to the number of tokens that can be input to GPT-3. Our preliminary testing showed that the upper limit of tokens is too small to confirm the effectiveness of few-shot learning in GPT-3.

III. Patent Classification Using Few-shot Learning

The goal of our work is to collect target patents by performing two steps: (1) patent search using a search formula and (2) applying document classification to the search results, assuming a situation where 100–200 training cases are available. This section describes our approach in detail.

In this study, we use patents with manually annotated F-term codes for pretraining. F-term is represented by nine-digit alphanumeric characters, and is divided into two elements: the "theme" and the "viewpoint," which is subdivided into several classes such as the purpose of the invention, field of use, and materials.

In this study, we use two datasets, one for pretraining and the other for few-shot learning. There is no overlap in patents in each dataset, and patents containing F-term codes assigned by patents in the dataset for few-shot learning are excluded from the dataset for pretraining. We created two datasets: "random category" and "proximity category." Both categories use the F-term code assigned to the patent at the time of filing as a positive example. The random category randomly selects an F-term code that differs from the positive example and uses the patent containing that code as the negative example. The proximity category uses as negative examples patents that contain codes that are close to the F-term code of the positive example.

Figure 1 shows a part of the F-term code for the theme "powder metallurgy (4K018)." The differences in hierarchy are represented by the depth of indentation. In Figure 1, FA01 and FA08 are categories of the same depth, and these are adjacent categories. If a patent assigned FA01 is a positive example, then a patent assigned FA08 is selected as a negative example.

- FA01 Mechanical treatments
  - FA02 Sizing or coining
    - FA03 Processing in general or apparatus therefor
      - FA04 Lubrication or pre-treatments of sintered materials
  - FA05 Densification of surfaces; Rolling
  - FA06 Cutting or polishing
- FA08 Heat treatments

Fig. 1. A part of F-term codes (a theme of "powder metallurgy" (4K018))

As a document classification task, "proximity category" is more difficult than "random category", because documents in the closer category must be classified. The reason for dealing with these two types of data and creating several datasets with different conditions is that the actual tasks to be solved have a different difficulty of classification for each task. In addition, we also investigate different granular classifications, such as thematic and perspective levels.

The patents used in the dataset are Japanese patents. In this study, we use the abstract described in the patent with F-term codes assigned to each patent.

IV. Experiments

A. Experimental Conditions

Experimental Data

There are two main categories of experimental data: first, patents from 2004–2014 were used for pretraining and then patents after 2018 were used for few-shot learning (Table I). There is no overlap between the data for pretraining and the data for few-shot training, and there is no overlap between themes and F-terms. The documents to be input into the model are the abstract portions of the patents. Pretraining was performed on 287,971 training data, 191,981 validation data, and 36,679 test data.

TABLE I. Data Description

Years	Total Number of Patents	Number of Theme	Number of F-term
2004-2014 (Pretraining)	479,952	349	2,664
2018 (Few-shot)	141,093	264	2,209

In addition, from the data for few-shot learning, we created five types of datasets, each of which is a single topic group of 200 data, with 100 positive examples and 100 negative examples, as shown below.

[1] Dataset with random category (viewpoint level)
Dataset of 100 topics in total with F-term code at viewpoint level.

[2] Dataset with proximity category (viewpoint level)
Dataset of 68 topics in total with F-term code at viewpoint level.

[3] Dataset with 2-digit proximity category (between theme and viewpoint level)
Dataset of 17 topics in total with 2-digit proximity F-term code. The first two digits of the F-term code are "FA" in the case of Figure 1, and the task is to determine whether the patent belongs to the "FA" category.

[4] Dataset with random category (theme level)
Dataset of 30 topics in total, with F-term code at theme level.

[5] PatentNoiseFilter Dataset
The PatentNoiseFilter[1] dataset is a small-scale patent classification dataset used in actual practice in PatentNoiseFilter, a patent classification system provided by Hatsumei-Tsushin Co., Ltd. The following three topics are contained in this dataset.

- Disposable Surgical Masks: Use a dataset on one topic with 150 patents related to disposable surgical

[1] https://www.hatsumei.co.jp/patentnoisefilter/

masks as positive data and 150 unrelated patents as negative data.

- Fishing Tackle: Collect patents related to reels, lures, fishing rods, and other fishing tackle, and use three types of data: (1) reels for positive data and the others for negative data; (2) reels and lures for positive data and the others for negative data; and (3) reels, lures, and fishing rods for positive data and the others for negative data, with 100 positive and 100 negative samples.
- Surgical Masks: Five patents related to surgical masks were extracted for the positive data and five unrelated patents for the negative data, divided into learning, evaluation, and testing (3:1:1) to create a dataset of 10 ways ($_5C_3$).

Experimental Method

First, for few-shot learning, we built a classifier to classify F-terms using about 500,000 Japanese patents. BERT (cl-tohoku/bert-base-japanese-whole-word-masking) was used as the base model. Based on this model, we built models for each F-term using few-shot learning. This model achieved a macro-average F1 value of 0.305 when performing classification on F-terms.

In addition to few-shot learning, we examined BERT [8], SVM, and random forest (RF) as comparative machine learning methods. BERT is a pretrained model that uses a large corpus with a two-way encoded representation by transformer as pretraining data and is a model that has achieved high precision in tasks such as document classification. The few-shot learning model is based on BERT pretrained on patent data. SVM is a type of machine learning model that determines the support vectors and a certain straight line with the maximum distance, and performs various tasks such as text classification. RF is an ensemble learning algorithm that uses multiple decision trees, that is, a machine learning technique that uses tree structures to perform classification. In this study, we use tf-idf to extract feature value of documents for SVM and RF. Evaluation is based on Precision, Recall, and F1 value. The threshold for each label is 0.5.

B. Experimental Results

The experimental results are shown in Tables II–VI. For random F-term, Few-shot produced the highest results in Precision, Recall, and F1 value. In contrast, for the experiments using proximity data, Few-shot fell below SVM in one part for the 7-digit proximity F-term and the proximity F-term. For the PatentNoiseFilter dataset, Few-shot produced the best results in terms of Precision, Recall, and F1 value.

TABLE II. RESULTS OF DATASET WITH RANDOM CATEGORY (VIEWPOINT LEVEL)

Method	Precision	Recall	F1 value
Few-shot	**0.944**	**0.940**	**0.942**
BERT	0.888	0.860	0.874
SVM	0.914	0.925	0.919
RF	0.913	0.910	0.911

TABLE III. RESULTS OF DATASET WITH PROXIMITY CATEGORY (VIEWPOINT LEVEL)

Method	Precision	Recall	F1 value
Few-shot	**0.693**	0.680	**0.686**
BERT	0.601	0.592	0.596
SVM	0.668	**0.688**	0.678
RF	0.672	0.663	0.667

TABLE IV. RESULTS OF DATASET WITH 2-DIGIT PROXIMITY CATEGORY (BETWEEN THEME AND VIEWPOINT LEVEL)

Method	Precision	Recall	F1 value
Few-shot	0.745	**0.737**	0.741
BERT	0.665	0.643	0.654
SVM	**0.790**	0.724	**0.756**
RF	0.763	0.726	0.744

TABLE V. RESILTS OF DATASET WITH RANDOM CATEGORY (THEME LEVEL)

Method	Precision	Recall	F1 value
Few-shot	**0.908**	**0.905**	**0.906**
BERT	0.781	0.749	0.765
SVM	**0.908**	0.900	0.904
RF	0.847	0.838	0.842

TABLE VI. RESULTS OF PATENTNOISEFILTER

Method	Precision	Recall	F1 value
Few-shot	**0.787**	**0.792**	**0.783**
BERT	0.655	0.715	0.672
SVM	0.758	0.680	0.660
RF	0.756	0.710	0.716

V. DISCUSSION

A. Overall Experimental Results

From the random category dataset in Table II and the proximity category dataset in Tables III–V, we see that all evaluation indices are higher in the case of Few-shot than BERT. This is likely because pretraining of the patent data plays an important role in classifying patents by successfully acquiring a vector of patent-related terminology. In contrast, BERT, which was not pretrained, was not able to acquire that vector well, and its F1 value is lower than that of Few-shot.

B. Dataset with Random Categories

Table II shows that all evaluation indices are higher for Few-shot than for SVM and RF. This is likely because of Few-shot's acquisition of patent word vectors through pretraining of patent data and BERT's unique understanding of the context of sentences, both of which contribute to the improvement in precision. In contrast, SVM and RF use tf-idf, which does not take context into account, to acquire word vectors; therefore, the quality of the word vectors is poor, and the F1 value is lower than that of Few-shot.

C. Dataset with Proximity Categories

Tables III–V show that the values of evaluation indices become smaller in the order of random category (theme level), 2-digit with proximity category, and proximity category (viewpoint level). This is because it becomes more difficult to distinguish categories as the granularity of patent classification becomes finer. Table V shows that the results of Few-shot, SVM, and RF do not significantly differ, but only BERT is lower. Tf-idf, which is used in SVM and RF to extract feature vectors, can deal with unknown words, but it does not consider word order. However, in the case of theme-level classification, for example, the question is "whether a patent is machine-translated or not", so even without considering word order, classification can be performed if there are words related to machine translation. Therefore, in the case of SVM and RF, the values of the evaluation indices are high. On the other hand, in the case of BERT, although the order of words is considered, the values of the evaluation indices are not high. From the above, patent classification without pretraining about patents is more likely to improve the classification result by corresponding to unknown words rather than to word order. However, as the difficulty of classification increases, such as the 2-digit level and viewpoint level, the values of the evaluation indices are almost the same between BERT, SVM, and RF. This is because the granularity of classification becomes finer, such as "whether the composition of a certain ingredient is greater than or less than [numerical number]" as a classification criterion to separate adjacent F-terms, and the word order becomes more important. Therefore, when the difficulty of classification increases, it is necessary to consider the word order as in BERT.

D. *Dataset with Patent-separated Categories*

SVM showed excellent classification performance for proximity categories at viewpoint level, for 2-digit proximity categories, and for random categories at theme level. On the other hand, as for the PatentNoiseFilter dataset, Recall was found to be extremely poor for some topics. From the user's perspective, stability is important. Even if the average value is high, if it becomes extremely low depending on the search task, the system cannot be used. Thus, we evaluated the experimental results of the PatentNoiseFilter dataset using the GMAP evaluation scale.

$$GMAP = \exp\left(\frac{1}{n}\sum_{i=0}^{n} ln\ x_i\right) \qquad (1)$$

This is a measure of the robustness of the system and is calculated as the geometric mean, not the arithmetic mean, of the Precision, Recall, and F1 value for each search task. By using multiplication, the GMAP value will drop sharply if any one of them contains an extremely low value.

TABLE VII. RESULTS OF PATENTNOISEFILTER DATASET WITH GMAP

Method	Precision	Recall	F1 value
Few-shot	**0.768**	**0.779**	**0.768**
BERT	0.600	0.699	0.636
SVM	0.747	0.597	0.626
RF	0.742	0.694	0.705

The results show that Few-shot is superior to the other methods not only in terms of the average values shown in Table VII, but also in terms of robustness.

E. *Few-shot Code Prediction Errors in the Proximity F-term Dataset*

A common thread of error was the absence of words in the patent's abstract that characterized the F-term to which it belonged. In the case of neighboring F-term, often only certain partial words differ, and the other contents are almost the same. Therefore, if there is no word that characterizes the F-term to which it belongs, it becomes difficult to classify it correctly. In fact, in cases of correct classification, the abstract of patents often contains important words. This suggests that it is necessary to increase the amount of information by using claims and other information, rather than using only the abstract.

VI. CONCLUSION

In this study, a dataset was created using approximately 600,000 patents, and a patent classifier using few-shot learning was built. The experimental results showed that among the five datasets based on F-term, the patent classifier based on the proposed method—few-shot learning—was the best in almost all cases. In addition, in the experiments using the PatentNoiseFilter dataset, Few-shot was the best for macro-averages of Precision, Recall, and F1 value and GMAP.

REFERENCES

[1] S. J. Pan, and Q. Yang: A Survey on Transfer Learning, IEEE Transactions on Knowledge and Data Engineering, Vol. 22, Issue 10, 2010, pp. 1345-1359.

[2] J. Snell, K. Swersky, and R. Zemel: Prototypical Networks for Few-shot Learning, Proceedings of the 31st Conference on Neural Information Processing Systems, 2017.

[3] W. Chen, Y. Liu, Z. Fira, Y. F. Wand, J. Huang: A Closer Look at Few-shot Classification, Proceedings of the 7th International Conference on Learning Representations, 2019.

[4] C. Finn, P. Abbeel, and S. Levine: Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks, Proceedings of the 34th International Conference on Machine Learning, 2017, pp. 1126-1135.

[5] X. Li, H. Chen, Z. Zhang, and J. Li: Automatic Patent Classification using Citation Network Information: An Experimental Study in Nanotechnology, Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, 2007, pp 419-427.

[6] M. Yu, X. Guo, J. Yi, S. Chang, S. Potdar, Y. Cheng, G. Tesauro, H. Wang, and B. Zhuo: Diverse Few-shot Text Classification with Multiple Metrics, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1206-1215, 2018.

[7] T. Brown et al.: Language Models are Few-shot Learners, Proceedings of the 34th Conference on Neural Information Processing Systems, 2020.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova: ERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171-4186, 2019.

[9] A.H. Roudsari, J. Afshar, W. Lee, and S. Lee, PatentNet: Multi-label Classification of Patent Documents Using Deep Learning Based Language Understanding, Scientometrics, Vol. 127, pp. 207-231, 2022.

[10] P. Tang, M. Jiang, B.N. Xia, J.W. Pitera, J. Welser, and N.V. Chawla, Multi-Label Patent Categorization with Non-Local Attention-Based Graph Convolutional Network, AAAI Technical Track: Natural Language Processing, Vol. 34, 2020.