https://doi.org/10.5392/IJoC.2017.13.2.066

Text Classification for Patents: Experiments with Unigrams, Bigrams and Different Weighting Methods

ChanJong Im
Department of Information and Telecommunication
Pai Chai University, Daejeon, 155-40, Republic of Korea
University of Hildesheim, Hildesheim, 31141, Germany

DoWan Kim
Pai Chai University, Daejeon, 155-40, Republic of Korea

Thomas Mandl
University of Hildesheim, Hildesheim, 31141, Germany

ABSTRACT

Patent classification is becoming more critical as patent filings have been increasing over the years. Despite comprehensive studies in the area, there remain several issues in classifying patents on IPC hierarchical levels. Not only structural complexity but also shortage of patents in the lower level of the hierarchy causes the decline in classification performance. Therefore, we propose a new method of classification based on different criteria that are categories defined by the domain's experts mentioned in trend analysis reports, i.e. Patent Landscape Report (PLR). Several experiments were conducted with the purpose of identifying type of features and weighting methods that lead to the best classification performance using Support Vector Machine (SVM). Two types of features (noun and noun phrases) and five different weighting schemes (TF-idf, TF-rf, TF-icf, TF-icf-based, and TF-idcef-based) were experimented on.

Key words: *Data Mining, Patent Classification, Information Retrieval, Machine Learning, Support Vector Machine (SVM), Supervised Weighting Scheme, Noun Phrase Extraction, Bag of Words.*

1. INTRODUCTION

With the increasing importance of Research & Development (R&D) based on the technological development, Intellectual Property (IP) became one of the vital subjects to be analyzed. Especially in the industry, IP R&D, which refers to the analysis of patents and research papers of the competing companies, is becoming one of the fundamental methods in building R & D strategies [17].

Along with the growing importance of IP R&D, the number of patent application filings have increased dramatically and keep growing over the years [2], [27]. To allow the users to have easy and fast access to the needed patent information, International Patent Classification (IPC) is used as a classification system. This system is revised annually and reformed occasionally to keep up with the newly developed technologies. However, such regular updates make it extremely difficult for the patent examiners to keep up with the changes and to classify the patent documents in accordance with these changes. It is not only difficult to classify the vast amount of newly filed patents, i.e. pre-classification, but also to rearrange and update the previously classified documents into the modified classification, i.e. reclassification [2]. Moreover, the classification carried out on the lower levels of the IPC hierarchy showed its limitation due to the insufficient number of documents in the dataset. In order to reduce the labor of examiners and the cost for both pre-classification and reclassification, automatic patent document classification with precise text classification algorithms is needed. In addition, alternative sources of categories need to be use in order to increase the performance.

Unlike most of other classification works conducted on the patent domain, we propose the usage of technical categorization explicitly shown on the Patent Landscape Reports (PLR) which are provided by World Intellectual Property Organization [32]. PLR are trend analysis reports which are written by the domain's experts. The PLR provide categories for the domains which are composed of classes and subclasses. Using these categories will not only reduce the existing complexity in the classification system leading to the

** Corresponding author, Email: johnjamin11@naver.com*

Manuscript received Jun. 09, 2017; revised Jun. 21, 2017; accepted Jun. 21, 2017

performance increase but also support the construction of value added systems, e.g. trend mining, patent retrieval systems.

Text classification (TC) is a supervised learning tasks where the documents containing texts are classified into a predefined set of classes automatically [20]. Multi-label classification refers to the automatic classification into three or more classes. Prominent algorithms used for text classification are Naïve Bayes and Support Vector Machine (SVM).

In the field of information retrieval (IR), natural language processing (NLP), and other context related areas, various studies have used the Bag-of-Words (BoW) approach as a basis for constructing representation schemes and preprocessing. A number of studies were conducted which dealt with different weighting methods [7], [20], [31] and different features [4], [6].

In this paper, we will carry out several classification experiments on the categorization levels obtained from PLR with five weighting schemes (Tf-idf, TF-rf, TF-icf, TF-icf-based, and TF-idcef-based) and two feature types (noun and noun phrases). Throughout the paper, the experiments carried out on class level will be referred to as Classification task 1 and those carried out on subclass level will be referred to as Classification task 2.

2. METHODOLOGY

In this section, there will be a description about theories and resources that are related to this work. In Section 2.1, categories, classes, and subclasses from the PLR and the queries used for retrieval will be explained. In Section 2.2 and following subsections, there will be descriptions related to data preparation. In Section 2.3, classification algorithms along with the related software will be illustrated.

2.1 Patent Landscape Reports and golden standard queries

Enabling fast access to the required information is one of the biggest reason for the existence of classification. The most basic way to retrieve relevant information is by using keywords search. However, it is very difficult to obtain relevant patents by using keyword search. The languages used in patents is extremely complicated and the terms are purposely used in a vague way to allow the widest coverage in the domain [8].

Another way to obtain relevant patents is to use the predefined classification codes, e.g. IPC, [10]. Unfortunately, a study by [29] pointed out that using datasets retrieved by the queries which were solely built with predefined classification codes may not meet the goal of particular analysis, i.e. trend analysis, when dealing with patents. Due to the complexity of IPC or CPC, it is extremely difficult and time consuming to search for a specific classification code, especially for the users who do not have any knowledge of the domain.

In order to have comprehensive and easy access to the required information in the patent domain, both keywords and a predefined classification scheme should be used when searching. Such queries are found in the trend analysis reports, i.e. PLR.

The World Intellectual Property Organization (WIPO) and their partner organizations such as international organizations, national intellectual offices, non-governmental organizations, and private sector entities regularly publish Patent Landscape Reports (PLR). PLR give users a general overview of the current status or trends in a specific field or technology. The reports contain analyses conducted on specific countries or regions and global overviews. These reports help decision makers in the industry to make strategic decisions [1].

Some of the reports explicitly show the queries that were used in their "*Patent Search Strategy*". These are constructed by the experts of the domain for each arbitrarily defined domain and category. The queries are complex and long which are composed of two parts, i.e. main and sub-queries.

Main queries are constructed with keywords of the domain and combinations of predefined classification. For example, in an analysis report on '*contact lenses*' [14], the main query contains the keyword query "*contact* wd3 lens**" and a combination of IPC and CPC codes related to the domain. When the main queries are used in the patent search, all the patents related to contact lenses are given in return. The classification carried out on the outcomes that result from the main queries is called *class level classification*.

Sub-queries are built only with keywords. These keywords are chosen by the domain experts based on the subclasses in predefined categories. For example, in the same '*contact lenses*' report, the experts have technically segmented the domain into three categories which are "*Types", "Materials",* and *"Use".* A number of subclasses are allocated inside each category. In category *"Use",* for instance, there are four subclasses *"Astigmatism", "Hyperopia", "Myopia",* and *"Presbyopia".* The keywords used in each sub-query are chosen based on these subclasses. The outcome returned from the search using sub-queries only contains patents that are related to the subclasses.

2.2 Data preparation

In this section, related works on data preparation will be elaborated on. In Subsection 2.2.1, a description of the Vector Space Model and the Bag of Words approach will be given. In the following Subsections 2.2.2 and 2.2.3, different kinds of weighting methods and English phrases used in representation will be explained respectively.

2.2.1 Bag of Words (Vector Space Model): Texts in patents need to be modified into a state that machine can classify the documents. The conventional way of representing text is by using the Bag of Words (BoW) method, also referred to as Vector Space Model (VSM). BoW does not consider the semantics or the order of the text. It considers all the terms in the documents as features of vectors. Each feature, corresponding to each term, is weighted using number of weighting methods.

$$d_j = (w_{1,j}, w_{2,j}, \dots\ w_{i,j}) \qquad (1)$$

Equation 1 shows the equation for the document-term matrix where d_j represents the j^{th} document and $w_{i,j}$ represents the i^{th} term in the j^{th} document.

2.2.2 Weighting methods: [20] pointed out three factors for term weighting assignment which have been discussed in the field of information retrieval.

The first factor is the term frequency factor which comes from the idea that the number of occurrence of a term (*tf*) in the document is related to the content of the document. It is referred to as a local variable as it is representing the information within the document. The most basic forms of term frequency factors include *Binary, tf, LogTF,* and Inverse Term frequency (*ITF*).

The second factor is the collection frequency factor which considers discriminative powers between the documents. It relies on the idea that the terms which are not common in the dataset have the highest discriminative power. This factor is also referred to as a global factor as it considers the frequency in the whole document collection. One of the most well-known examples is Inverse Document Frequency (*idf*).

For text classification tasks, where labels for each category are assigned, various term weighting methods use information of categories instead of global factors. This is known as supervised term weighting. Table 1 shows elements of supervised term weighting where a set of categories *C* with *k* as the number of categories is defined as $\mathrm{C} = (c_1, c_2, \dots, c_k)$. The supervised term weighting method is commonly used in various kinds of multi-label classification tasks [7].

Table 1. Elements of supervised term weighting [7]

	c_k	$\bar{c_k}$
t_i	A	C
$\bar{t_i}$	B	D

According to the article [7], each element in Table 1 is defined as the following:

- A: number of documents belonging to category c_k which contain t_i at least once.
- B: number of documents belonging to category c_k that does not contain t_i.
- C: number of documents not belonging to category c_k that contain t_i at least once.
- D: number of documents not belonging to category c_k that does not contain t_i.
- N: total number of documents which is equivalent to $|\mathrm{D}| = \mathrm{A} + \mathrm{B} + \mathrm{C} + \mathrm{D}$.

Using these notations, *idf* is defined as shown in the following equation 2.

$$idf(t_i) = log\frac{N}{A + C} \qquad (2)$$

One example of supervised global weighting scheme is Inverse Category Frequency (*icf*).

$$icf(t_i) = log\frac{|C|}{|c\colon t_i \in c_x|} \qquad (3)$$

Equation 3 shows the equation of *icf*. It is the logarithm of $|\mathrm{C}|$ which is total number of categories, divided by $|c\colon t_i \in c_x|$ which refers to the number of categories containing t_i. Compared to *idf* where the global information of the terms in the document collection is used, *icf* uses global information of the terms in the categories. Similar to *idf*, the intuition behind *icf* is that the terms which occur in a fewer categories have more discriminative power in the classification tasks [31].

Another supervised term weighting scheme that was introduced by [20] is Relevance Frequency (*rf*). The main idea comes from the name itself that it only considers relevant documents that contain the term. In other words, the more documents contain t_i in the positive category, the more influential it is when selecting the positive documents from the negative documents. Their work showed that *rf* outperformed other supervised weighting schemes such as *logOR*, X^2, *ig* in text classification tasks. The equation is shown in the following:

$$rf(t_i) = \log(2 + \frac{A}{Max(1, C)}) \qquad (4)$$

[31] proposed a new weighting scheme called *icf-based* due to the fact that *rf* \ lead to lower discriminating power in certain circumstances since it only considers frequency of relevant documents. The equation is shown in the following:

$$icf - based(t_i) = \log(2 + \frac{A}{Max(1, C)} * \frac{|C|}{|c\colon t_i \in c_x|}) \qquad (5)$$

Icf-based showed better results in classification tasks than other supervised weighting schemes. Especially when it was used on the imbalanced dataset, the performance increased substantially.

Recently, [7] proposed a new weighting scheme called Inverse Docment Frequency Excluding Category (*idfec-b*). It adds more weights to the terms that are more discriminative than the common ones. The equation is shown in the following:

$$idfec - b(t_i) = \log(2 + \frac{A + C + D}{Max(1, C)}) \qquad (6)$$

All three weighting methods, i.e. equations 4, 5, 6, were tested on Reuters-52 and 20 Newsgroup datasets with SVM classifier. Reuters-52 is imbalanced dataset whereas 20 Newsgroup is a dataset which is uniformly distributed across the categories. The best overall performance was seen when tested with sufficient size of feature using *rf*. However, *idfec-b* outperformed *rf* when the experiments were held using an imbalanced dataset and a relatively small size of features.

The last factor for the term weighting assignment is normalization factor. It comes from the idea that long documents which contain more terms can be not as important as the short documents with less terms. Therefore, to eliminate the length effect, cosine normalization is generally used. The following is the equation for cosine normalization.

$$w_{k,j}norm = \frac{w_{k,j}}{\sqrt{\sum_1^k (w_{k,j})^2}}) \qquad (7)$$

2.2.2 English phrases used for representation: The types of features that are used in text classification have been studied since the early 90s. Attempts to improve classification results by adding features of n-grams were conducted. According to an overview of the development by [6], there were various attempts to combine unigrams with statistical phrases [4] and unigrams with syntactical phrases [22]. The result shows for both statistical and syntactical phrases that representation using single n-gram reduced the accuracy of the classification task. However, the increase in the performance was shown when unigrams and bigrams were used together. Articles [18] and [19] showed patent classification using triples retrieved by the dependency parser. The best performance was shown when the triples were used with unigrams.

The article by [6] showed that the characteristics of the text are affective in classification accuracy. Patents contain more noun phrases compared to the normal text, e.g. Reuters collection. In their patent classification experiments, the good performance results were shown when the mixture of syntactical bigram phrases obtained from the dependency parser and unigrams were used as features. They also conducted a subsidiary survey to find out the most influential feature types in patents. They asked the experts to select the most important phrases in patents and, as a result, noun-noun compounds and adjectival modifier with a noun were chosen. The results from the experiments and the survey imply that noun phrases in patents are the most influential feature type in the classification performance.

2.3 Classification

There have been several software tools which were developed for data processing and classification. One of the well-known tools based on Java is WEKA [15]. It provides various kinds of algorithms for both data processing and classification. In particular for classification, it supports number of machine learning models such as Naïve Bayes, Neuron Networks, K-Nearest Neighbor, and SVM.

Among the provided models, SVM [26] is one of the most widely used supervised classifier in text classification. It is a binary classifier which aims to find the separating hyperplane with the maximum margin. It is known to be simple, fast and efficient especially for big sized data with many features [20]. The research by [23] has claimed that using linear kernel is more efficient than using other kernel types for text classification. There were only a small performance differences among the kernel types but the linear one was the fastest in processing and computing the data. Two SVM models are provided in WEKA which are SMO [24] and LibSVM [5].

Choosing the best training model is one of the most important things to consider when conducting classification tasks. The best model is known to be obtained with the classifier that maintains the balance between over fitting and under fitting, also known as bias and variance. *K-folds* cross validation which is provided by WEKA is a one of the way to keep the balance between the two.

Since SVM is a binary classifier, there were several studies which found a way to use this classifier in multiclass classification tasks [16]. Among several approaches, OAA (One-Against-All) and OAO (One-Against-One) are the methods supported by WEKA. For k number of classes, OAA constructs k classifier and OAO constructs $k(k-1)/2$ classifiers. Although OAO method constructs more classifier than OAA, OAO was recommended for practical use due to the speed and complexity of computation.

There are several ways to evaluate classification performances. The simplest way to assess is by calculating overall classification accuracy. This can be obtained by dividing total number of instances from total number of correctly classified instances. Although this measure intuitively shows how well the classification tasks were progressed, it is not widely used for the evaluation since it leaves out the details of small segments, i.e. evaluation for each class.

The prominently used measurement is F_1 measures which are the harmonic average of precision and recall. There are two kinds of F_1 measurements: *Macro*-F_1 and *Micro*-F_1. *Macro*-F_1 is a measurement which was proven to be effective in small sized classes and *Micro*- F_1 was proven to be effective in evaluating large sized classes [30]. The following show the equations for *Macro*-$F_1$and *Micro*-F_1 [3]:

$$Macro - F_1 = 2 * \frac{\overline{p_x} * \overline{r_x}}{\overline{p_x} + \overline{r_x}} \qquad (8)$$

$$Micro - F_1 = 2 * \frac{p^g * r^g}{p^g + r^g} \qquad (9)$$

The differences between the two measures in equation 8 and 9 are the precision (*p*) and recall (*r*) values. For $Macro - F_1$, precision mean values ($\overline{p_x}$) for all classes *x* and recall mean values ($\overline{r_x}$) for all classes *x* are calculated from each confusion matrix derived from cross validation. Global precision (p^g) and global recall (r^g) values are used in $Micro - F_1$ which are calculated from a single pooled confusion matrix from cross validation.

3. EXPERIMENTS

All the procedures involved in patent classification contain four big phases: data collection preparation, preprocessing and indexing, vector generation, and classification. Each phase will be elaborated in the following sections respectively.

3.1 Data collection preparation

There are total of three sets of data prepared the classification tasks, i.e. Classification task 1 (balanced and imbalanced) and Classification task 2.

Dataset for task 1 is prepared for the experiments on the class level. Not all patent domains are selected for the experiment. We chose to select the ones which contained queries explicitly shown in the analysis reports. As a result, three different patent domains are selected: "*Contact Lenses*", "Slot Machine", and "Robotic Arms" from each corresponding PLR [12]-[14]. Using the software established in advance [11], all queries are passed onto the European Patent Office (EPO) database. Table 2 shows the total number of patents obtained for each class.

Table 2. Number of documents for Task no.1

Class	Total	Indexed
Contact Lenses	267	250
Robotic Arms	296	250
Slot Machine	751	250

To compare the effectiveness of the different weighting schemes, two datasets are prepared. First is a balanced dataset which contains an equal number of documents in each class. In the indexing process, the number of documents are limited to size of 250 for each class as shown in Table 2 in the Indexed column. Second is the imbalanced dataset. The total numbers of documents obtained from querying are contained in this dataset. The numbers are shown in Total column in Table 2.

The dataset for task 2 is prepared on the subclass level. Similar to task 1, not all subclasses are selected for the experiments. We tried to classify the documents with the subclasses placed in a category mentioned in the PLR for a specific domain "*Contact Lenses*". As mentioned in Section 2.1, the domain is semantically segmented into three categories: "*Types*", "*Materials*", and "*Use*". We chose the subclasses placed within the category "*Use*" which is "*Astigmatism*", "*Glaucoma*", "*Hyperopia*", "*Keratoconus*", "*Myopia*", and "*Presbyopia*". However, the number of patents placed in "*Keratoconus*" and "*Glaucoma*" was too small to be used in the experiments. Thus, these subclasses were excluded from the dataset. The final dataset that are used in the experiments is shown in Table 3.

Table 3. Number of documents for Task no.2

Subclass	Number of Patents
Astigmatism	166
Hyperopia	321
Myopia	199
Presbyopia	65

3.2 Preprocessing and indexing

Four steps are required to increase the classification performance in the preprocessing phase. These steps include delimiter removal, noun phrase extraction, stemming, and stop word elimination.

It is important to remove unnecessary characters contained in the retrieved patents. Unneeded characters are stored in 'delimiters' variable and they are removed from the text using the WordTokenizer provided by WEKA.

The whole text in the patent documents is not efficient for the classification tasks. As it was emphasized by [6], noun phrases are the most important features in patent classification. Various types of nouns (unigrams) and noun phrases (bigrams) were recommended. The list of unigrams and bigrams that were used for extraction is shown in Table 4.

Table 4. Type of nouns used as features

POS Tags	Description
NN	Noun
NNS	Noun, plural
NNP	Proper noun
NNPS	Proper noun, plural
DT + NN	Determiner + noun
NN + NN	Noun + noun
JJ + NN	Adjective + noun
JJ + JJ	Adjective + Adjective
RB + JJ	Adverb + Adjective

In addition, unigrams and bigrams are not extracted from the whole text. It was shown by [2] that using text in the abstract section in patents is highly efficient and effective. However, we chose to use title and claim sections due to the assumption that more noun phrases are located in the title and claim sections than they are in the abstracts.

For the actual extraction, we chose to use Part of Speech Tagging [28] and pattern detection [21]. Extraction using a dependency parser was not considered due to the fact that dependency parsers were not initially made for patent text processing. Patents contain typically very long sentences which are hardly processed by the current technology.

Table 5. Combinations types and number of documents for task no.2

Combination types	Number documents
P	0
H	96
A	4
M	5
P∩H∩A∩M	237
P∩H∩A	6
P∩H∩M	9
P∩A∩M	0
H∩A∩M	204
P∩M	0
P∩A	0
P∩H	0
H∩M	125
H∩A	63
A∩M	2

Stemming and stop word elimination are the last steps in preprocessing. Stemming is used for reducing the words to their stem which ultimately leads to the reduction of the dimensions in document vectors. A widely used tool for stemming, the Porter Stemmer, is used [25]. Stop word elimination takes place after stemming and just before the features are indexed. The elimination is based on the idea that extremely frequent words do not contain any information. However, some terms are useful but have the high frequencies. In the case of processing dataset extracted from class '*Contact*

Lenses', for example, the term 'lens' occurs very frequently in the collection. Despite its high frequency, the term has high discriminative power which is rarely found in other classes. Thus, we chose to remove only the general list of stop words obtained from the work by University of Neuchatel [9]. Note that stop word elimination is not implemented in bigram extraction for POS tagging accuracy.

To allow efficient control over all the extracted features, all the terms are indexed using the Apache Lucene API. Note that for classification task 1, labels for each class are given during this process.

For classification task 2, additional steps are required for document labeling due to the duplication problem. Since documents on the subclass level were extracted by using the keywords search, a problem occurs in the documents which belong to more than one subclass. In other words, some documents contain more than one class labels.

To resolve the problem, additional categories are created so that each document could be assigned to just one category. Instead of using the four subclasses mentioned in Table 3, a new set of categories $C = \{c_1, c_2, c_3, \ldots, c_{|c|}\}$ is created where $|c|$ correspond to total number of category combinations. The total number of combinations of categories is defined in the following equation 10 where *n* denotes total number of subclasses.

$$|C| = \sum_{r=1}^{n} \mathrm{nCr} = \sum_{r=1}^{n} \frac{n!}{r!\,(n-r)!} \qquad (10)$$

When equation 10 is implemented to classification task 2, the total number of subclass categories becomes 15. These newly established categories and numbers of documents named after each category are shown in Table 5. Note that the abbreviations for subclasses shown in the table refer to the following: Presbyopia = 'P', Hyperopia = 'H', Astigmatism = 'A', Myopia = 'M'.

3.3 Vector generation

Table 6. Number of documents and unigram features

	Task no.1	**Task no.2**
Total number of docs	750	751
Total number of features	3549	2858

Table 7. Number of documents and bigram features

	Task no.1	**Task no.2**
Total number of docs	750	751
Total number of features	15116	7538

All documents vectors are generated from the indexed data in the vector generation phase. The dimensions in each vector correspond to the weights calculated with the weighting schemes mentioned in Subsection 2.2.2.

Table 6 and 7 show the total number of documents and features for both classification tasks 1 and 2 with unigrams and bigrams respectively. The values placed on the rows named "Total number of docs" represent *j* documents used for each corresponding task on the column. The values placed on the rows named "Total number of features" refer to the total size of the vector dimensions (*i+1*) for each corresponding task. For example, the size of the dimensions for all *j* documents for classification task 1 with bigrams is equivalent to 15116. Note that one dimension is added to each document vectors for document labels assignment.

3.4 Classification

After all document vectors are formed, they are then used for classification experiments using the tool WEKA. LibSVM with linear kernels are used to construct training models and 10-folds cross validation is carried out to evaluate the performance. Note that for classification task 1 balanced and imbalanced datasets are used for the classification experiments and imbalanced dataset with newly established labels is used in classification task 2.

4. RESULTS

The results for both classification task 1 and 2 will be shown in this section. We used *Macro* and *Micro* F_1measures for the evaluation. More specifically, both *Micro* and *Macro* F_1 measures are used for assessing experiments carried out with imbalanced dataset and *Micro*-F_1 is used for evaluating the experiments carried out with balanced dataset.

4.1 Experiment results for classification task 1

Table 8. Classification results for task no.1 (balanced)

	TFidf	TFicf	TFrf	TFicf-based	TFidfec-b
Unigrams					
Micro-F_1	0.975	0.975	0.988	**0.990**	0.988
Bigrams					
Micro-F_1	0.963	0.975	0.985	0.988	0.976

The main purpose of classification task 1 was classifying the patents which were retrieved on the class level. Two datasets, i.e. balanced and imbalanced were prepared. Table 8 shows the experiment results which used the balanced dataset with two types of features (unigram and bigram) and five weighting methods.

As written in bold in Table 8, the best performance is shown when unigrams and the TF-icf-based weighting scheme are used. For both unigram and bigram, the lowest performance is shown when TF-idf is used. This implies that supervised weighting schemes outperform the conventional weighting scheme when used in classification tasks. When comparing the results specifically between the feature types, classification using unigrams performed better than classification using bigrams. Though the differences between the results are not that remarkable, it implies that using more features as noted in Tables 6 and 7 does not lead to better classification results.

Also, it implies that bigrams are less discriminative than the unigrams when using a balanced dataset in classification task 1.

Table 9. Classification results for task no.1 (imbalanced)

	TFidf	TFicf	TFrf	TFicf-based	TFidfec-b
Unigrams					
Micro-F_1	0.973	0.974	**0.991**	**0.991**	0.986
Macro-F_1	0.971	0.973	**0.991**	**0.991**	0.985
Bigrams					
Micro-F_1	0.940	0.952	0.983	0.986	0.963
Macro-F_1	0.937	0.950	0.982	0.986	0.962

Table 9 shows the results for the experiments carried out using the imbalanced dataset. The best results are achieved in the experiment carried out with feature of unigrams and weighting scheme of TF-rf and TFicf-based. When comparing the results among different weighting schemes, TFicf-based experiments show the best performance in both types of features. When comparing the results among types of features, unigram outperformed bigram with small difference. Both results in Table 8 and 9 show that classification carried out on the class level with unigrams and TFicf-based weighting scheme lead to the highest F_1 scores.

4.2 Experiment results for classification task 2

The main purpose of classification task 2 was to classify the patents which were retrieved on the subclass level. The imbalanced dataset containing ten classes from combination allocation was prepared and used for the experiments. The experiment results are shown on Table 10.

Table 10. Classification results for task no.1 (balanced)

	TFidf	TFicf	TFrf	TFicf-based	TFidfec-b
Unigrams					
Micro-F_1	0.907	0.889	0.889	0.941	0.907
Macro-F_1	0.639	0.632	0.631	0.662	0.639
Bigrams					
Micro-F_1	0.940	0.949	0.943	**0.961**	0.943
Macro-F_1	0.664	0.671	0.666	**0.681**	0.666

As written in bold in Table 10, the best performance is achieved when bigrams and TFicf-based weighting scheme are used. The results of the experiment using TFicf-based outperformed other experiment results in all cases among the weighting schemes results.

It is important to note that TF-idf outperforms some of the supervised learning schemes. This implies that despite the fact that TF-idf does not consider the category information, it still possesses high discriminative power especially in the case when the categories used are constructed from combination allocation. In other words, not all supervised weighting schemes are highly discriminative when dataset with similar features are used.

The experiments using bigrams show better results than when unigrams are used. This implies that bigrams containing diversified features have a higher effectiveness in the task. Since the dataset used in task 2 was labeled with the duplicates combinations of subclasses, the set of features between the duplicates and the origin class share similar features to a great extent. It is difficult to distinguish the combination set from the origin set by using unigrams with a relatively small set of features.

Different from the results in Table 9, significant gaps between the values of *Micro* and *Macro* measures are observed in Table 10. This implies that the instances are classified well for the larger classes but not well for the smaller classes.

5. CONCLUSION AND FUTURE WORK

With the increasing interest in Intellectual Property (IP) and the increasing amount of patent filings, the need for classification programs with high accuracy have been augmented. Various classification studies were conducted using the datasets retrieved on IPC hierarchy levels. However, some problems still remain when using the IPCs. They are too broad making classification tasks tedious and time consuming. Also, the prior works showed the limitations in classifying the patent documents on the low IPC hierarchical levels.

This paper implemented different kinds of hierarchy levels for classification tasks. Instead of using IPCs, the hierarchical categories as well as the queries obtained from the PLR were used. Although the classifications were not carried out on the IPCs, doing classification on the dataset retrieved through using the queries containing full-IPCs, CPC and keywords gave an effect on narrowing down the range of broad IPCs.

The experiments using these categories showed high performing results. For classification task 1, which was carried out on class levels, the best results were obtained when TFicf-based and unigrams were used for the experiments using both balanced and imbalanced datasets. For classification task 2 where the duplicates combination categories were used, the best results were shown when TFicf-based and bigrams were used.

This work suggests the usage of categories other than IPC or CPC. As shown in the results, the work shows the potential to be extended into value added systems such as trend mining and patent retrieval systems.

To make any improvements on the experiment results, four things need to be reconsidered. First is the size of the datasets. The larger the dataset the better is the performance. More patents could be obtained from other international patent offices. However, the language differences are the problematic factor when processing the documents. To resolve the language problem multi-lingual patent retrieval system or multi-lingual processing tools need to be developed.

Second thing to be considered is the way the documents are labeled. In classification task 2, the documents were labeled based on how the documents were distributed over the subclasses. All possible combinations from the subclasses were considered as labels. This labeling method not only reduced the number of documents in each label making some subclasses unable to be classified but also increased the complexity of computation. Alternative methods need to be experimented on.

The third thing to be considered is the type of features. In classification task 1, using unigrams led to better performance than bigrams. However, using bigrams led to better results in the second task. This implies that classes share similar feature sets which leads to a high misclassification rate in classification task 1. In our work, the experiment was not carried out using both unigrams and bigrams as one feature set. Thus, different types of features will be examined.

The last thing to be considered is weighting schemes especially for classification task 2. When using the same document labeling method mentioned on this paper, the weighting scheme that could give more weights to the distinctive features contained in one class is needed.

REFERENCES

[1] T. Anthony, *Guidelines for preparing patent landscape reports*, http://www.wipo.int/edocs/pubdocs/en/wipo_pub_946.pdf, 2015. [Last accessed: 20th of May 2016].

[2] K. Benzineb and J. Guyot, “Automated patent classification, in ‘Current challenges in patent information retrieval’,” Springer Berlin Heidelberg, 2011, pp. 239-261.

[3] C. Bielza, G. Li, and P. Larranaga, “Multi-dimensional classification with Bayesian networks,” International Journal of Approximate Reasoning, vol. 52, no. 6, 2011, pp. 705-727.

[4] M. F. Caropreso, S. Matwin, and F. Sebastiani, “Statistical phrases in automated text categorization,” Centre National de la Recherche Scientifique, Paris, France, 2000.

[5] C. C. Chang and C. J. Lin, “Libsvm: a library for support vector machines,” ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, 2011, p. 27.

[6] E. D’hondt, S. Verberne, C, Koster, and L. Boves, “Text representations for patent classification,” Computational Linguistics, vol. 39, no. 3, 2013, pp. 755-775.

[7] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori, “A study on term weighting for text categorization: a novel supervised variant of tf. idf,” in Proceedings of the 4th international conference on data management technologies and applications (DATA). Candidate to the best conference paper award, 2015, pp. 26-37.

[8] D. Eisinger, G. Tsatsaronis, M. Bundschus, U. Wieneke, and M. Schroeder, “Automated patent categorization and guided patent search using ipc as inspired by mesh and pubmed,” Journal of biomedical semantics, vol. 4, no. 1, 2013, p. 1.

[9] University of Neuchatel: Stop word list, online, http://members.unine.ch/jacques.savoy/clef/index.html.,2005- [Last accessed: 28.10.2016].

[10] EPO and USPTO, “Guide to the CPC,” http://www.cooperativepatentclassification.org/publications/GuideToTheCPC.pdf, 2015. [Last accessed: 20th of May 2016].

[11] N. Fadaei, T. Mandl, M. Schwantner, M. Sofean, J. M. Struß, K. Werner, and C. Womser-Hacker, “Patent analysis and patent clustering for technology trend mining, in ‘Elbes hausen Stefanie, Faaÿ Gertrud, Griesbaum Joachim, Heuwing Ben, Jürgens Julia (Hrsg.), HIER 2015 - Proceedings des 9. Hildesheimer Evaluierungs- und Retrieval workshop,” Hildesheim University Hildesheim, pp. 77-86, 2015 [Last accessed: 5th of May 2016].

[12] Grid Logics Technologies, “Technology insight report slot machines,” http://www.patentinsightpro.com/techreports/0511/Technology%20Insight%20ReportSlot%20Machines.pdf, 2011. [Last accessed: 4th of June 2016].

[13] Grid Logics Technologies, “Technology insight report robotic arms,” http://www.patentinsightpro.com/techreports/0312/Robotic%20Arms%20Tech%20Report.pdf, 2012. [Last accessed: 4th of June 2016].

[14] Grid Logics Technologies, “Contact lenses technology insight report,” http://www.patentinsightpro.com/techreports/0514/Tech%20Insight%20Report%20%20Contact%20Lens.pdf, 2014. [Last accessed: 22th of May 2016].

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The weka data mining software: an update,” ACM SIGKDD explorations newsletter, vol. 11, no. 1, 2009, pp. 10-18.

[16] C. W. Hsu and C. J. Lin, “A comparison of methods for multiclass support vector machines,” IEEE transactions on Neural Networks, vol. 13, no. 2, 2002, pp. 415-425.

[17] S. Ik Jae, *Patent management strategy using hierarchical clustering*, Master's thesis, Korea University, Korea, 2014.

[18] C. H. Koster and J. G. Beney, “Phrase-based document categorization revisited,” Proceedings of the 2nd international workshop on Patent information retrieval, ACM, 2009, pp. 49-56.

[19] C. H. Koster, J. G. Beney, S. Verberne, and M. Vogel, “Phrase-based document categorization, in ‘Current challenges in patent information retrieval,” Springer, 2011, pp. 263-286.

[20] M. Lan, C. L. Tan, J. Su, and Y. Lu, “Supervised and traditional term weighting methods for automatic text categorization,” IEEE transactions on pattern analysis and machine intelligence, vol. 31, no. 4, 2009, pp. 721-735.

[21] N. Oleg, *Eigennamenerkennung für technologien. implementierung und evaluierung eines prototyps für patente*, Master's thesis, University of Hildesheim, Germany, 2016.

[22] L. Özgür and T. Güngör, “Text classification with the support of pruned dependency patterns,” Pattern Recognition Letters, vol. 31, no. 12, 2010, pp. 1598-1607.

[23] C. Park, D. Seong, and K. Lee, “Automatic ipc classification for patent documents using machine learning,” Journal of Korean Institute of Information Technology, vol. 10, no. 4, 2012, pp. 119-128.

[24] J. Platt, et al., *Sequential minimal optimization: A fast algorithm for training support vector machines*, 1998.

[25] M. F. Porter, “Snowball: A language for stemming algorithms,” http://snowball.tartarus. org/texts/introduction.html.[Last accessed: 4th of June 2016], 2001.

[26] A. Shmilovici, *Support vector machines, in ‘Data Mining and Knowledge Discovery Handbook*, Springer, 2005, pp. 257-276.

[27] J. M. Struß, T. Mandl, M. Schwantner, and C. Womser-Hacker, “Understanding trends in the patent domain,” in ‘IPaMin@ KONVENS’. http://ceur-ws.org/Vol-1292/ipamin2014_paper9.pdf,2014. [Last accessed: 16th of November 2016].

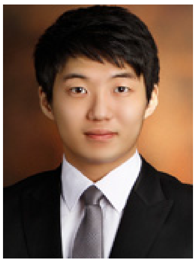
[28] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, “Feature-rich part-of-speech tagging with a cyclic dependency network,” Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, Association for Computational Linguistics, 2003, pp. 173-180.

[29] Y. J. Tseng, C. J. Lin, and Y. I. Lin, “Text mining techniques for patent analysis,” Information Processing & Management, vol. 43, no. 5, 2007, pp. 1216-1247.

[30] V. Van Asch, “Macro-and micro-averaged evaluation measures [[basic draft]]”. http:// scholar.google.co.kr/scholar?hl=ko&q=Macro-and+micro-averaged+evaluation+measures&btnG=&lr=, 2013. [Last accessed: 31th of October].

[31] D. Wang and H. Zhang, “Inverse-category-frequency based supervised term weighting schemes for text categorization,” Journal of Information Science and Engineering, vol. 29, no. 2, 2013, pp. 209-225.

[32] WIPO, “Patent Landscape Reports,” http://www.wipo.int/patentscope/en/programs/patent_landscapes/, 2016. [Last accessed: 16th of November].



ChanJong Im

He received B.S. in International Business from PaiChai University, Korea in 2014. He received M.A in Information Science from University of Hildesheim, Germany, and PaiChai University, Korea in 2016. He is currently doing his PhD at University of Hildesheim. His main research interests are in information retrieval, data mining, natural language processing, machine learning, and deep learning.

DoWan Kim

He received the B.A., M.A. and Ph.D. in Informatics from University of Regensburg, Germany. He was senior researcher in ETRI. Since 1997, he is professor at PaiChai University. His research interests include semantic web technologies, artificial intelligence, software quality evaluation and software ergonomics.

Thomas Mandl

He is professor for Information Science at the University of Hildesheim in Germany where he is teaching within the program International Information Management. He studied Information Science and Computer Science at the University of Regensburg in Germany, the University of Koblenz and at the University of Illinois at Champaign/Urbana, USA. He first worked as a research assistant at the Social Science Information Centre in Bonn, Germany. He received both a doctorate degree in 2000 and a post doctorate degree (Habilitation) in 2006 from the University of Hildesheim. His research interests include information retrieval, human-computer interaction and internationalization of information technology.