# Index Based Approach for Text Categorization

Taeho Jo and Dongho Cho
KAIST Institute for IT Convergence
335 Gwahangno Yusong Gu Daejon, 305-701
SOUTH KOREA
tjo@kaist.ac.kr, dhcho@ee.kaist.ac.kr

***Abstract*—** This research proposes an alternative approach to machine learning based approaches for categorizing online news articles. For using machine learning based approaches for any task of text mining, documents should be encoded into numerical vectors; it causes two problems: huge dimensionality and sparse distribution. Although there are various tasks of text mining such as text categorization, text clustering, and text summarization, the scope of this research is restricted to text categorization. The idea of this research is to avoid the two problems by encoding a document or documents into a table, instead of numerical vectors. Therefore, the goal of this research is to develop a scheme which is free from the two problems for categorizing on-line news article automatically.

***Keywords*—** Text Categorization Index-based Approach, Machine Learning based Approach, and Text Mining.

## I. Introduction

Text categorization is the process of assigning one or some among predefined categories to each document. The task belongs to pattern classification where texts or documents are given as patterns. Note that almost information in any system is given as textual formats dominantly over numerical one. For managing efficiently the kind of information given as the textual format, techniques of text categorization are necessary; 'text categorization' became a very interesting research topic in both academic and industrial worlds. However, as the preprocessing, documents or texts should be encoded into numerical vectors for using traditional techniques for the task.

Encoding documents so causes the two main problems. The first problem is huge dimensionality where documents must be encoded into very large dimensional numerical vectors. In general, documents must be encoded at least into several hundreds dimensional numerical vectors in previous literatures [1][3][4]. This problem causes very high costs for processing each numerical vector representing a document in terms of time and system resources. Much more training examples are required proportionally to the dimension for avoiding over-fitting.

The second problem is sparse distribution where each numerical vector has zero values dominantly. In other words, more than 90% of its elements are zero values in each numerical vector. This problem degrades the discrimination among numerical vectors. This causes poor performance of text categorization. In order to improve performance of text categorization, the two problems should be solved.

The idea of this research is to avoid the two problems by encoding documents into tables instead of numerical vectors. The proposed approach to text categorization is called index based class of approaches in this research. Each table is a collection of entries consisting of words and their weights indicating the importance of words in a given document or a corpus. Category by category, we can sum weights of matched words between a table given as the surrogate of a document and a table as a categorical profile which will be explained later. Therefore, an unseen document is classified as the category corresponding to the maximum summed weight.

The performance of the proposed approach will be validated through experiments in section V. For doing that, the test beds used for the experiments are a particular collection of news articles: NewsPage.com. The proposed approach is compared with the two machine learning based approaches: KNN (K Nearest Neighbor) and NB (Naïve Bayes). F1 measure where recall and precision are combined with their equal proportion is adopted as the evaluation measure. In section V, it is shown that once the optimal option is given, the proposed approach is better than any machine learning based approach.

This paper consists of six sections including introduction. In section II, we will survey previous cases of applying one of the machine learning based approaches to text categorization. In section III and 4, we will describe the process of encoding documents into tables and the proposed text categorization system, respectively. In section V, the performance of the proposed approach is validated by comparing the approach with the two machine learning based ones on the test bed. In section VI, we will mention the significance of this research and further research as the conclusion.

## II. Previous Works

This section concerns the exploration for previous research on text categorization. In 2002, Sebastiani mentioned two kinds of approaches to text categorization in his research paper [9]. One is rule based class of approaches and the other is machine learning based one of approaches. He did not consider the former since the class of approaches is very naïve, and he surveyed only machine learning based class. Among approaches belonging to the machine learning based class, we

This work is supported by Brain Korea 21 Project, the school of Information Technology, KAIST, in 2008

will survey four representative approaches: KNN (K Nearest Neighbor), NB (Naïve Bayes), SVM (Support Vector Machine), and Neural Networks in this section, because of their popularity.

The first representative approach to text categorization is KNN. In 1992, KNN was initially applied to the classification of news articles by Massand et al [6]. In 1999, Yang analyzed 12 approaches to text categorization with each other, and observed through her experiments that KNN is one of recommendable approaches [11]. In 2002, Sebatiani evaluated KNN as a simple and competitive algorithm with SVM which was evaluated as the best algorithm. Its disadvantage is that KNN costs very much time for classifying objects, given a large number of training examples because it must compute similarities of each unseen example for all individual training examples to select some of them.

Another popular approach to text categorization is NB. This approach is a variant of the Bayes Classifier based on the Bayesian Rule which assumes the independence of attributes [7]. In 1997, Mitchell mentioned NB as a typical approach to text categorization in his text book [7]. In terms of a supervised learning algorithm, its advantage is that it learns training examples with its higher speed than neural networks. However, its disadvantage is that an almost zero value of probability influences on the entire posteriori probability; a smoothing scheme was proposed for solving the problem [7].

The third representative approach to text categorization is SVM. In 1998, it was initially applied to text categorization by Joachims [4]. He validated the better classification performance of SVM in text categorization by comparing it with KNN and NB. Drucker et. al. adopted SVM for implementing a spam mail filtering system and compared it with NB in implementing the spam mail filtering system in 1999 [3]. In 2000, Cristianini and Shawe-Taylor presented a case of applying SVM to text categorization in their textbook [2].

The last representative approach to text categorization is Neural Networks. Among models of neural networks, MLP (Multi Layers Perceptron) with the back propagation algorithm is most popular model . The model of neural networks was initially applied to text categorization in 1995 by Wiener [10]. In 2002, Ruiz and Srinivasan applied several MLPs to text categorization by combining them hierarchically [8]. The combined model of neural networks in their research was called HME (Hierarchical Mixture of Experts).

In order to apply one of traditional machine learning based approaches including the four representative approaches, documents must be encoded into numerical vectors. Encoding so causes the two main problems: huge dimensionality and sparse distribution as mentioned in section I. There was a previous attempt to solve the two problems without encoding documents so. In 2002, Lodhi et al proposed a string kernel for applying Support Vector Machine to text categorization [5], and in their research, documents are used as their raw form. However, their proposed version of SVM failed to be better than the traditional version of SVM [5].

## III. Document Encoding

This section concerns the process of encoding a document or documents into a table. The process described in this section implements the module named ‘text indexer’, in the architecture of the proposed text categorization system, shown in Fig. 3. A document or documents are given as input of the process as shown in Fig. 1. The process generates a table consisting of entries each of which consists of a word and its weight which indicates how much important the word is. The three schemes of weighting words will be also mentioned in this section.

Fig. 1 illustrates the process of encoding a document or documents into a table. If more than two documents are given as the input, their full texts are concatenated into a full text. A full text becomes the target for the tokenization. The full text is segmented into tokens by a white space or a punctuation mark in the first phase, ‘concatenation & tokenization’. Therefore, the first phase generates a list of tokens as its output.

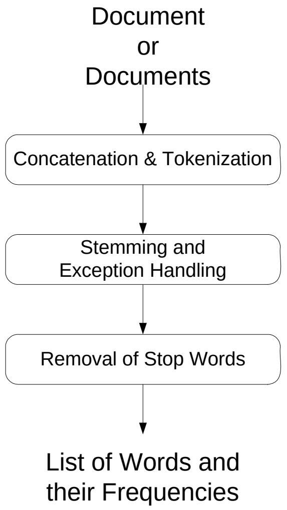

Fig. 1. Process of extracting a list of words and their frequencies

The output of the first phase is transferred to the next phase, ‘stemming and exception handling’ to its input. In this step, each token is converted into its root form. Before doing that, the rules of stemming and exception handling are saved into a file. When the program which encodes documents is activated, all rules are loaded into memory and the corresponding one among them is applied to each token. The output of this step is a list of tokens converted into their root forms.

The last step of extracting a list of words and their frequencies from a document or documents is to remove stop words as illustrated in Fig. 1. Here, stop words are defined as words which perform only grammatical functions irrelevantly to content of their document; articles (a an, or the), prepositions (in, on, into, or at), pronoun (he, she, I, or me), and conjunctions (and, or, but, and so on) belong to this kind of words. It is necessary to remove the kind of words for more efficient processing. After removing stop words, frequencies of remaining words are counted. Therefore, a list of words and their frequencies are generated from the process illustrated in Fig. 1.

Although there are other schemes of weighting words, we will mention only three representative schemes. Firstly, we can assign binary values to words as their weights; each binary value indicates whether its corresponding word is important or not. Secondly, the frequencies of words may be given as their weights by themselves. Thirldy, we can weight words using equation (1),

$$weight_i(w_k) = tf_i(w_k)(\log_2 D - \log_2 df(w_k) + 1) \quad (1)$$

where $weight_i(w_k)$ indicates a weight of the word $w_k$, indicates its content based importance in the document $i$, $tf_i(w_k)$ indicates the frequency of the word in the given corpus, $w_k$ in the document $i$, $df(w_k)$ is the number of documents including the word in the given corpus $w_k$, and $D$ is the total number of documents in the given corpus. Among the three schemes, we adopt the third one for computing weights of words in this research.

## IV. Proposed Text Categorization System

This section concerns the proposed text categorization in terms of its architecture and flow. Fig. 2 illustrates the architecture of the proposed text categorization. As shown in Fig. 2, there are two modules involved in the system. The first module is named as 'document indexer', and encodes a document or documents into a list of words and their frequencies. The second module is named as 'classifier', and categorizes directly an unseen document.

The left part of Fig. 2 shows the process of building category profiles using labeled sample documents. In the view of machine learning, the process may be called 'learning' [7]. tables are generated from this process and become references for categorizing unseen documents. Each table corresponds to each category. The weights of the table indicate the relevancy of words for the given category, and they are called categorical weights in this research.

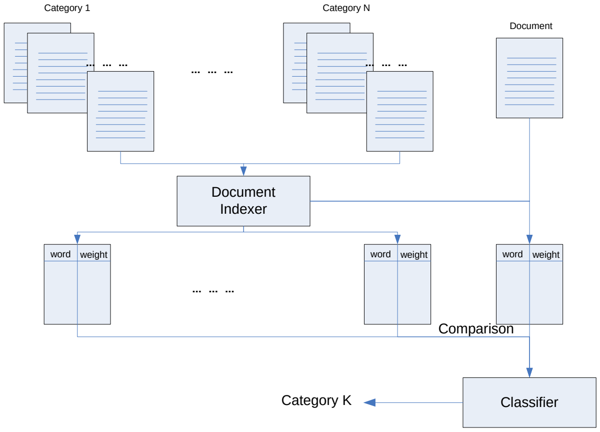

Fig. 2. System architecture of text categorization

The right part of Fig. 2 shows the process of categorizing an unseen document based on categorical weights. In this process, a particular unseen document is given as the input. The process corresponds to generalization in view of machine learning [7]. The unseen document is converted into a table by the module, 'document indexer'. The weights of the table given as the surrogate of the unseen document indicate the relevancy of words to its content; the weights are called substantial weights in this research.

We already above mentioned the two kinds of weights of words involved in the proposed text categorization system. One is categorical weights given in the categorical profiles. In this research, we may define the categorical weights as the real value indicating how much words are relevant to the given category. The other is substantial weights given in a surrogate of an unseen document. In this research, we define substantial weights which indicates how much words are relevant to the content of the document.

In the proposed system, unseen documents are categorized based on matched words between two tables. One is given as a surrogate of an unseen document, and the other is given as a categorical profile. We can get matched words from the two tables. We can compute a categorical score by summing the weights of the matched words. We will assign the category corresponding to the maximum categorical score to the unseen document.

## V. Empirical Results & Discussions

These experiments concern the comparisons of the two classes of approaches to text categorization. The first class of approaches is a class of machine learning based approaches, where documents should be represented into numerical vectors. Typical approaches belonging to this class are SVM, NB, and KNN. The second class of approach is an index based approach as the proposed one where documents are represented into a table of words, whose entry consists of a word and its weight. In the proposed approach, we define the two types of weights of words: substantial weights and categorical weights. A substantial weight of a word indicates how much it is important in the given document with respect to its content. Substantial weights of words are computed, when a document is indexed. A categorical weight of a word indicates how much it is relevant to the category. Categorical weights of words are computed, when documents belonging to their identical category are indexed.

The goal of this research is to compare the two classes of approaches to text categorization each other and validate that the index-based approach is more practical than the machine learning based approaches. In these experiments, we selected NB, SVM, and KNN as representative machine learning based approaches, since they are main popular and traditional approaches to text categorization. The proposed approach has the four options. In these experiments, we will observe the performance of the two classes approaches to text categorization, using two test beds: NewsPage.com, and Reuter21578.

The first test set for evaluating the two classes of approaches is NewsPage.com. This test bed consists of 1,200 news articles in the format of plain texts built by copying and pasting news articles in the web site, www.newspage.com. Table 1 shows the predefined categories, the number of documents of each category, and the partition of the test bed into training set and test set. As shown in table 1, the ratio of training set to test set is set as 7:3. Here, this test bed is called Newspage.com, based on the web site, given as its source.

**Table 1. Training Set and Test Set of Newspage.com**

| Category Name | Training Set | Test Set | #Document |
|---|---|---|---|
| Business | 280 | 120 | 400 |
| Health | 140 | 60 | 200 |
| Law | 70 | 30 | 100 |
| Internet | 210 | 90 | 300 |
| Sports | 140 | 60 | 200 |
| Total | 840 | 360 | 1200 |

The last test set for evaluating the two classes of approaches to text categorization is Reuter21578, which is a typical standard test bed in the field of text categorization. In this experiment set, most frequent ten categories are selected. Table 2 shows ten selected categories and the number of training documents and test documents in each category. The partition of the test bed into training set and test set follows the version, ModApte, which is the standard partition of Reuter 21578 for evaluating text classifiers. The number of documents in each category is very variable as shown in table 2. In this test set, since each document has more than one category, we evaluate the two classes of approaches to text categorization by decomposing it into ten binary classification tasks. In each binary classification task, each classifier generates ‘belonging’ or ‘not belonging’, as its output, instead of one of predefined categories. In this test set, one more machine learning based approach, SVM, participate in this evaluation, and the evaluation measure is determined as F1 measure, instead of accuracy.

As mentioned above, we select SVM, NB, and KNN as the representative machine learning based approaches for the comparison with the proposed approach. The selected machine learning algorithms have been used previously and popularly not only for text categorization, but also for any other pattern classification. Since SVM is applicable only to binary classification, it participates only in this test bed, Reuter21578, where text categorization is decomposed into multiple binary classification tasks. The parameter of SVM is capacity and its value is set to four through tuning. The parameter of KNN is K which indicates the number of retrieved sample labeled documents, and its value is set to three.

**Table 2. Partition of Training Set and Test Set in 20NewsGroup**

| Category Name | Training Set | Test Set | #Document |
|---|---|---|---|
| Acq | 1452 | 672 | 2124 |
| Corn | 152 | 57 | 209 |
| Crude | 328 | 203 | 531 |
| Earn | 2536 | 954 | 3490 |
| Grain | 361 | 162 | 523 |
| Interest | 296 | 135 | 431 |
| Money-Fx | 553 | 246 | 799 |
| Ship | 176 | 87 | 263 |
| Trade | 335 | 160 | 495 |
| Wheat | 173 | 76 | 249 |

We use the indexed based approach to text categorization with the four options as illustrated in table 3, in the three test beds. In the first option, categorical scores are computed based on the number of matching words as the base option. In the second option and the third option, categorical scores are computed by summing substantial weights and categorical weights of matching words, respectively. In the fourth option, categorical scores are computed by summing products of both weights of matching words.

**Table 3. Four Options in the Proposed Approach**

| | |
|---|---|
| First Option | Number of Matched Words |
| Second Option | Substantial Weights of Matching Words |
| Third Option | Categorical Weights of Matching Words |
| Fourth Option | Substantial * Categorical Weights of Matching Words |

Figure 3 illustrates the results of evaluating the proposed approach with the four options and the two machine learning based approaches on the first test bed, NewsPage.com. In figure 3, the left part shows the performance of the four options within the proposed approach, and the right part shows that of the two machine learning based approaches and the proposed approach with the third option. As illustrated in the left part of figure 3, the experiment on the first test set shows that the third option is the optimal option within the proposed approach. The right part of figure 3 shows that the proposed approach with the optimal option classifies documents more accurately than the two machine learning based approaches.

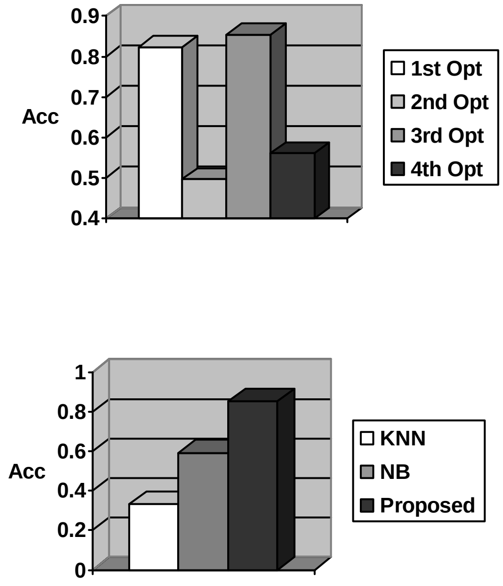

Fig 3. Results of Four Options of the Proposed Approach and ML based Approaches in NewsPage.com

Figure 4 shows the results of evaluating the three machine learning based approaches including SVM and the proposed one on the second test bed, Reuter 21578, which is the standard test bed. Since each document has one more than a label, we must apply the approaches decomposing text categorization into binary classification problems. So, we use F1 measure combining recall and precision, instead of accuracy, as an evaluation measure. There are two types of F1 measures: micro-averaged F1 and macro-averaged F1. Refer the textbooks on information retrieval and data mining for the detail explanation of the two types of F1. With respect to micro-averaged F1, the results on this test bed are identical to those on the previous test beds. However, with respect to macro-averaged F1, the results are somewhat different from those on the previous test beds; the third option is almost same as the first option within the proposed approach, and both KNN and the proposed method are identical to each other with respect to their performances.

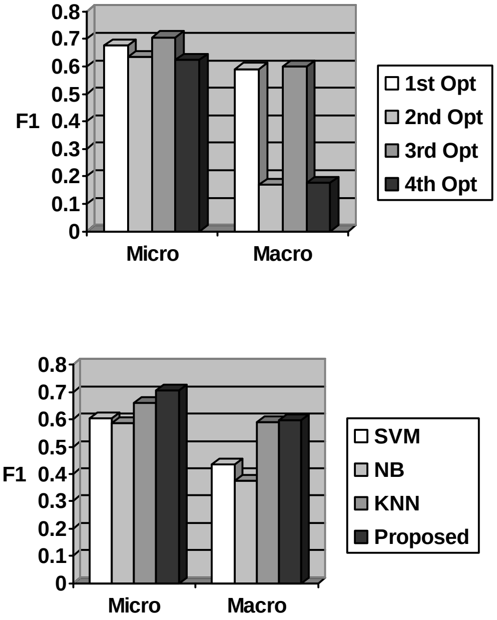

Fig 4. Results of Four Options of the Proposed Approach and ML based Approaches in Reuter21576

Let’s consider the four options within the proposed scheme. These experiments show that the third option where categorical scores are computed by summing categorical weights of matching words is validated as the best one in the three test beds. These experiments present that substantial weights are harmful for categorizing documents; the second option and the last option where substantial weights are considered for computing categorical scores got worse even than the first option set as the base option. In these experiments, we use sample labeled documents not only for building classification rules but also as a corpus for computing substantial weights. When we compute substantial weights of words in a particular document using the corpus, the weights of informative words for categorization are underestimated by the words included in other documents belonging to their identical topic. This underestimation of substantial weights of informative words degrades the performance of the proposed approach to text categorization.

It is concluded empirically from these experiments that the proposed approach with the third option classifies documents more accurately than the three machine learning based approaches. Documents should be encoded into numerical vectors for using machine learning based approaches such as SVM, NB, and KNN for text categorization tasks. This encoding leads to two main problems: huge dimensionality and sparse distribution. When documents are represented into sparse numerical vectors, classification performance is degraded, since discrimination among numerical vectors is lost. These experiments show this fact, empirically.

However, in these experiments we must consider the proposed method classifies documents better than machine learning based approaches only in the third option. The proposed approach with its second and last option is not good as the machine learning based approaches in all of test beds. As mentioned above, the reason is that documents belonging to an identical topic are regarded as different ones for computing substantial weights. It may be expected to address this problem if we use a separate corpus in a different domain, instead of a collection of training documents.

## VI. Conclusions

This research proposes an alternative approach to machine learning based approaches to text categorization. In the proposed approach, a document or documents are encoded into a table, instead of a numerical vector or numerical vectors. In other words, we can avoid the two main problems in encoding documents into numerical vectors: huge dimensionality and sparse distribution. The performance of the proposed approach was validated in the previous section. Since the two problems are solved, the proposed approach is shown to work better than machine learning based ones for text categorization.

There may be many ways of computing weights of words. In this research, we computed weights of words using equation (1), because of the popularity in the information retrieval. Note that the weights do not reflect exactly the relevancy of words to a given category or a content of a document. We need to develop several state of the art schemes for computing weights. In further research, we will compute weights of words by combining multiple schemes with each other.

If we could develop various schemes for computing weights of words, we may define multiple tables to a document or corpus. There are two ways for treating multiple tables. The first way is to integrate multiple tables corresponding to a document or a corpus into a table. The second way is to treat the multiple tables as a committee. In further research, we will evolve the proposed approach by encoding a document or corpus into multiple tables.

In this version of the proposed text categorization system, the number of entries of tables is fixed constantly. The proposed one is called static index based approach. However, the optimal number of entries is very dependent on the given document or corpus. The size of each table should be optimized in terms of two factors: reliability and efficiency. In the further research, we will propose dynamic index based approach where the size of table may be changed automatically with satisfying the both factors.

The weights of words may be automatically adjusted to improve the performance of text categorization in implementing the proposed approach. We need an additional set of labeled documents, called validation set. The set is built by separating some of a given training set from itself. The weights of words are updated to minimize misclassification rate of the examples in the validation set. The modified version may be regarded as a fusion of the proposed approach and the machine learning based one.

## ACKNOWLEDGMENT

This work is supported by Brain Korea 21 Project, the school of Information Technology, KAIST, in 2008.

## REFERENCES

[1] I. Androutsopoulos, K. Koutsias, K. V. Chandrinos, and C. D. Spyropoulos, “An Experimental Comparison of Naïve Bayes and Keyword-based Anti-spam Filtering with personal email message”, *In Proc, 23rd ACM SIGIR*, 2000, pp160-167.

[2] N. Cristianini and J. Shawe-Taylor, *Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000, ch 12.

[3] H. Drucker, D. Wu and V. N. Vapnik, “Support Vector Machines for Spam Categorization”, *IEEE Transaction on Neural Networks*, Vol 10, No 5, pp. 1048-1054, May, 1999.

[4] T. Joachims, “Text Categorization with Support Vector Machines: Learning with many Relevant Features”, *In Proc. 10th European Conference on Machine Learning*, 1998, pp143-151.

[5] M. J. Kearns and U.V, Vazirani, *An Introduction to Computational Learning Theory,* MIT Press, Cambridge, Massachusetts, 1994.

[6] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, Text Classification with String Kernels, *Journal of Machine Learning Research*, Vol 2, No 2, pp419-444. March, 2002

[7] B. Massand, G. Linoff, and D. Waltz, “Classifying News Stories using Memory based Reasoning”, *In Proc. 15th ACM International Conference on Research and Development in Information Retrieval*, 1992, pp59-65.

[8] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997, ch 6.

[9] M.E. Ruiz and P. Srinivasan, “Hierarchical Text Categorization Using Neural Networks”, *Information Retrieval*, Vol 5, No 1, pp87-118, April, 2002.

[10] F. Sebastiani, “Machine Learning in Automated Text Categorization”, *ACM Computing Survey*, Vol 34, No 1, pp1-47, January, 2002.

[11] E.D. Wiener, “A Neural Network Approach to Topic Spotting in Text”, *The Thesis of Master of University of Colorado*, 1995.

[12] Y. Yang, “An evaluation of statistical approaches to text categorization”, *Information Retrieval*, Vol 1, No 1-2, pp67-88, August, 1999.

**Taeho Jo:** Taeho Jo received PhD degree from University of Ottawa in 2006. Currently, he works for IT Convergence for KAIST as a senior research scientist. He has submitted and published more than 100 research papers to journals and proceedings. His research interests are text mining, neural networks, machine learning, and information retrieval.