Using the K Nearest Neighbor Method and BM25 in the Patent Document Categorization Subtask at NTCIR-5

Masaki Murata
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
murata@nict.go.jp

Toshiyuki Kanamaru
Kyoto University
Yoshida-Nihonmatsu, Sakyo, Kyoto 606-8501, Japan
kanamaru@hi.h.kyoto-u.ac.jp

Tamotsu Shirado
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
shirado@nict.go.jp

Hitoshi Isahara
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
isahara@nict.go.jp

Abstract

Patent processing is extremely important for industry, business, and law. We participated in the F-term categorization subtask at NTCIR-5, in which, we classified patent documents into their F-terms using the k-nearest neighbor method. For document classification, F-term categories are both very precise and useful. We entered five systems in the F-term categorization subtask. They obtained the best f-measures of all 18 participating systems. This confirmed the effectiveness of our method. After the contest, we performed the experiments again during the theme categorization subtask, even though we did not officially enter. The results showed that our system obtained higher f-measures than the highest obtained by the other systems performing the subtask. This also confirmed the effectiveness of our method.

Keywords: *Classification, patent documents, K nearest neighbor method, BM25*

1 Introduction

Patent processing is important in various fields, such as industry, business, and law. We entered our systems into the F-term categorization subtask at NTCIR-5, in which we classified the patent documents into their F-terms using the k-nearest neighbor method. The F-term categories are very precise and thus extremely useful for classifying patent documents. Furthermore this method is able to classify a large number of documents, which would be difficult using sophisticated machine learning methods, such as the support vector machine [1] and the maximum entropy methods [8], because these methods are complicated and require a lot of time and machine resources (memory). In contrast, the k-nearest neighbor method is comparatively easy to use for large amounts of data, because it only has to extract a set of data similar to the input data. Moreover, as Yang pointed out, the support vector machine and the k-nearest neighbor method are the best machine learning methods for document classification [12]. Therefore, we used the k-nearest neighbor method in this study.

2 Problem setting

In this section, we describe the problem handled in this study.

We participated in the F-term categorization subtask during the NTCIR-5 Patent Workshop [7], because F-term categories are very precise and extremely useful for classifying patents. In the subtask, we

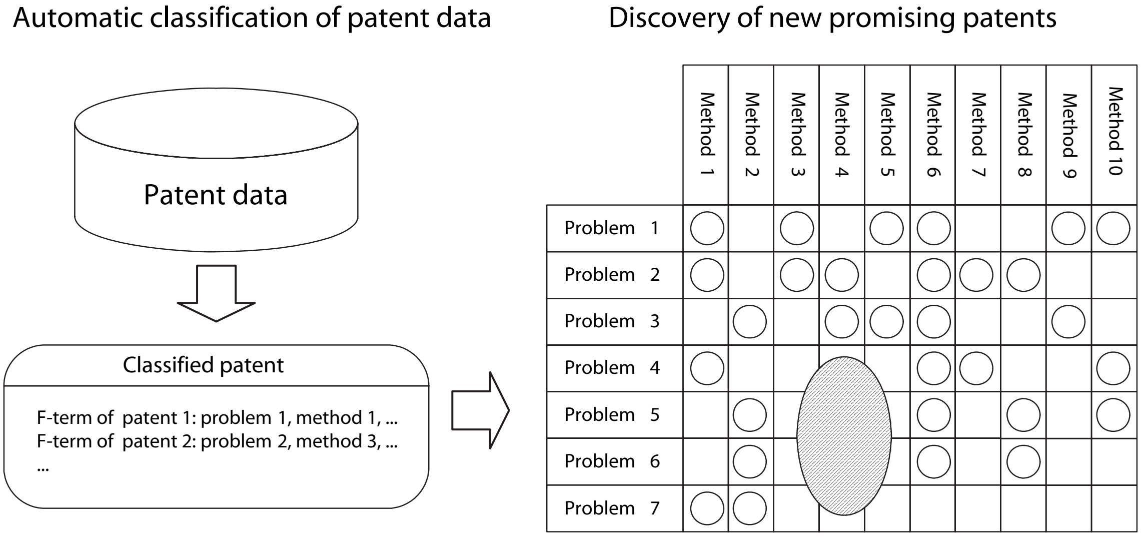

Figure 1. An example of using an F-term

determined the F-term categories of input Japanese patents when the category theme was given. Our problem was determining these categories. The NTCIR-5 Patent Workshop had a different subtask, classifying the themes. Although we did not participate in this subtask, we performed our experiments using the data collected in it. In the subtask, we determined the theme categories of the input patents. The subtask details are described on the NTCIR-5 Patent Workshop [7] website.

Each patent belongs to some theme categories and some F-term categories. Theme categories are a higher layer than F-term categories. Theme categories and F-term categories were added to each patent by the Japan Patent Office [4], and fell into about 2,600 theme categories, each with from dozens to thousands of F-term categories. Each patent had an average of 1.7 theme categories and 15 F-term categories in the formal run data.

The subtasks included a dry run and a formal run. For the F-term categorization subtask, we were given 1,201 patent documents to classify and 586,197 patent documents for training in the dry run. In the formal run, we were given 2,562 patent documents to classify and 1,508,043 patent documents for training. However, we could use documents with given theme categories for training. In the dry run, we were given about 8,000 patent documents with given theme categories. In the formal run, we were given 2,000 to 6,000 patent documents with given theme categories. For the theme categorization subtask, we were given 1,179 patent documents to classify and 586,197 patent documents for training in the dry run. In the formal run, we were given 2,009 patent documents to classify and 1,669,747 patent documents for training.

In the evaluation, we used average precision (A-Precision), r-precision, and f-measures. Average precision is the average of the precision when each category relevant to the input document is extracted. R-precision indicates the precision when extracting R categories, where R is the number of relevant categories. The f-measure is the average inverse of the recall and precision combined. The recall is the ratio of the correct outputs to all the correct categories. Precision is the ratio of the correct outputs to all the outputs.

3 Background and motivation

The F-term categories are both precise and useful for categorizing patents. For example, the “radio transmission” theme had many F-term categories, including“ purpose”, “application”, “transmission system”, “transmission signal”, “system architecture”, and “function”. These were then further broken down and “purpose” contained the F-terms of “failure prevention”, “service improvement”, and “efficiency improvement”; “application” contained the F-terms of “car phone”, “cellular phone”, and “train radio system”; and “function” contained the F-terms of “memorization”, “display”, etc. If we arrange the radio transmission patent documents into a two-dimensional table, where the columns are the “purpose” F-terms and the rows are the “application” F-terms, we can better understand the purpose and application situations in radio transmission patent documents. Figure 1 shows another simple example that demonstrates the usefulness of F-terms. In the example, each patent was given F-terms on problems and methods by automatically classifying the patent data. The patent information containing F-terms was transformed into the right side table in the figure. The circles in the table show that there are patents to handle the corresponding problems and methods. The part in the gray circle did not have any patents, which indicates the discovery of promising new patents, such as the patent handing problems 4 to 7 using methods 3 to 5. The F-terms are useful for discovering such patents. Thus, F-term categories can be very useful for categorizing patents. (The patent task organizer also illustrated the importance of F-term categorization for similar reasons.)

This study is useful for the following reasons:

- Our method can help annotators determine the F-term categories of each patent document.
- Our method can be used for documents from outside the patent office that do not contain F-term categories, and can assign F-term categories to these documents.

4 Variations of the k-nearest neighbor method

We used the following three variations of the k-nearest neighbor method.

1. Method 1

   The system first extracts the k patent documents with the highest similarities to an input patent document for all patent documents with the same given input theme in a training data set. We used the ruby-ir toolkit [10, 11] to extract the documents and experimentally determined the constant k.

   The system next calculates $Score$ using the following equation for each F-term category in the extracted documents.

   $$Score = \sum_i (k_r * score_{doc}(i)) \qquad (1)$$

   In this equation, k_r is the constant determined in the experiments, and $score_{doc}(i)$ is the similarity value of the ith extracted document.

   The system finally extracts the F-term categories with higher $Score$s than the highest $Score$ multiplied by k_p. We experimentally determined the constant k_p. The extracted F-term categories are output as the desired categories.

2. Method 2

   The system first extracts the k documents with the highest similarities to the input document for all the patent documents with the same given theme in a training data set.

   The system then extracts the F-term categories that occur more than k_t times in the k documents, where k_t is the product of k and k_u, and k_u is the experimentally determined constant. The extracted F-term categories are output as the desired categories.

   When $k_u = 0.5$, this method is exactly the same as the original k-nearest neighbor method.

3. Method 3

   The system first extracts the k documents with the highest similarities to the input document for all the patent documents with a given theme in a training data set.

   It then extracts the most frequently occurring k_a F-term categories from the extracted documents, where k_a is the average number of F-term categories each document has in the k extracted documents. The extracted F-term categories are output as the desired categories.

Methods 2 and 3 refer to and expand Lewis's k-per-doc and probability threshold strategies [5].

To test the effectiveness of each method, we also used the following baseline methods in the experiments.

1. Baseline 1

   The system first extracts all the F-term categories from all the patent documents with a given theme in a training data set. The system randomly extracts the k_b F-term categories as the desired output, where k_b is the average number of F-term categories of all documents with a given theme.

2. Baseline 2

   The system first extracts all the F-term categories from all patent documents with a given theme in a training data set, and sorts them in the order of the frequency of their appearance in the documents. The system extracts the most frequent k_b F-term categories from them as the desired output, where k_b is the average number of F-term categories of all documents of a given theme.

3. Original knn method [2]

   The system first extracts the k documents with the highest similarities to the input document from all the patent documents with a given theme in a training data set.

   The system then extracts the F-term categories that occur more than k_t times in the k documents, where k_t is the product of k, and k_u and k_u is 0.5. The extracted F-term categories are output as the desired categories.

5 Method of calculating similarity

We used the following two methods to calculate the similarity between an input patent document and each patent document in the training data set.

1. BM25

   The system first extracted terms [1] for each input patent document. It then extracted documents containing at least one of the terms, using the following equation to calculate $Sim_{BM25}$for each extracted document. We used Sim_{BM25} as the similarity between the input patent document and each patent document in the training data.

   $$Sim_{BM25} = \sum_{t \in T} (W_d \times W_q), \tag{2}$$

   where

   $$W_d = \frac{(k_1 + 1)tf}{k_1((1-b) + bdl/avdl)} \tag{3}$$

   $$W_q = log \frac{N - n + 0.5}{n + 0.5}. \tag{4}$$

   In these equations, T is the set of terms appearing in both the input and the extracted documents, tf is the number of occurrences of a term t in the extracted document, dl is the length of the extracted document, $avdl$ is the average length of the documents, N is the total number of documents, n is the number of extracted documents, andk_1 and b are the constants determined from the experiments. We used the default values described in the ruby-ir toolkit as k_1 and b ($k_1 = 1.5$ and $b = 0.75$).

   Robertson et al. proposed the BM25 algorithm [9], which is known to be a very accurate method of retrieving information [6].

2. Overlap

   The system first extracts terms for each input patent document. It next extracts documents containing at least one of the terms. The system uses the following equation to calculate $Sim_{Overlap}$ for each extracted document. We used $Sim_{Overlap}$ as the similarity between an input patent document and each document in the training data set.

   $$Sim_{Overlap} = |T|, \tag{5}$$

   where T is the set of terms appearing in both the input document and the extracted document, and $|T|$ means the number of members in the set T.

6 Regions used to extract terms

We extracted terms from the following three regions of the patent document.

1. Abstract

2. Claim

3. Domain and solution

   We used the first paragraphs in the "technical domain of the invention" and "method" regions in the patent documents, because they were related to the F-term categories.

7 Experiment

7.1 Experiments in the F-term categorization subtask

We first performed experiments during the F-term categorization subtask. We used variations of the k-nearest neighbor method, along with various other methods for calculating similarity. We extracted terms from all three regions (the abstract, claim, and domain and solution).

The results are shown in Table 1. We tested various values for each parameter and experimented using all combinations of these values to determine the best values for each parameter. The evaluation scores and parameter values that resulted in the highest scores for each method on the dry run data are shown in Table 1.

We also experimented using the formal run data by applying the parameter values that resulted in the highest scores for each method on the dry run data. The results are also shown in Table 2.

We used the dry run data to determine the parameters, and the formal run data to evaluate the effectiveness of the methods.

We used the two-sided t-test to determine significant differences, and the best methods (method 1 and BM25) as the baseline methods. The baseline methods were labeled "$*$". When a method performed better than the baseline method at the 0.05 or 0.01 significance level, it was labeled "$+$" or "$++$". Likewise, when a method performed worse than the baseline method at the 0.05 or 0.01 significance level, it was labeled "$-$" or "$--$".

Table 2 shows the following.

- When we compared the variations of the k-nearest neighbor method, method 1 had the best score. The difference between method 1 and the other three methods was very small (about 0.01). However, we confirmed that method 1 was more effective than the other methods using a statistical test at a significance level of 0.01.

[1] We only used nouns as terms.

Table 1. Experimental results from the F-term categorization dry run

Similarity method	Parameters	A-Precision	R-Precision	F-measure
Baseline 1				
		0.0383^{--}	0.0328^{--}	0.0324^{--}
Baseline 2				
		0.4484^{--}	0.4188^{--}	0.3991^{--}
kNN				
BM25	$k = 21$	0.5488^{--}	0.5093^{--}	0.3704^{--}
overlap	$k = 21$	0.5263^{--}	0.4776^{--}	0.3899^{--}
Method 1				
BM25	$k = 101, k_r = 1, k_p = 0.3$	0.5817*	0.5209*	0.5133*
overlap	$k = 101, k_r = 0.95, k_p = 0.3$	0.5436^{--}	0.4890^{--}	0.4797^{--}
Method 2				
BM25	$k = 101, k_u = 0.2$	0.5780^{--}	0.5180	0.5063^{--}
overlap	$k = 51, k_u = 0.3$	0.5441^{--}	0.4905^{--}	0.4784^{--}
Method 3				
BM25	$k = 51$	0.5749^{--}	0.5207	0.5043^{--}
overlap	$k = 51$	0.5441^{--}	0.4905^{--}	0.4723^{--}

Table 2. Experimental results from the F-term categorization formal run

Similarity method	Parameters	A-Precision	R-Precision	F-measure
Baseline 1				
		0.0597^{--}	0.0449^{--}	0.0396^{--}
Baseline 2				
		0.3306^{--}	0.3112^{--}	0.2962^{--}
kNN				
BM25	$k = 21$	0.4758^{--}	0.4548^{--}	0.2733^{--}
overlap	$k = 21$	0.4405^{--}	0.4162^{--}	0.2790^{--}
Method 1				
BM25	$k = 101, k_r = 1, k_p = 0.3$	0.5028*	0.4642*	0.4420*
overlap	$k = 101, k_r = 0.95, k_p = 0.3$	0.4612^{--}	0.4272^{--}	0.4083^{--}
Method 2				
BM25	$k = 101, k_u = 0.2$	0.4918^{--}	0.4534^{--}	0.4243^{--}
overlap	$k = 51, k_u = 0.3$	0.4538^{--}	0.4223^{--}	0.3636^{--}
Method 3				
BM25	$k = 51$	0.4942^{--}	0.4574^{--}	0.4330^{--}
overlap	$k = 51$	0.4538^{--}	0.4223^{--}	0.4022^{--}

Table 3. Results of formal run of NTCIR-5 Patent Workshop.

System	A-Precision	R-Precision	F-measure
Our system 1	0.4974	0.4563	0.4379
Our system 2	0.4728	0.4371	0.4190
Our system 3	0.4974	0.4563	0.4168
Our system 4	0.4974	0.4563	0.4258
Our system 5	0.4998	0.4611	0.4393
Team 1's system 1	0.1814	0.1950	0.1604
Team 1's system 2	0.1989	0.1994	0.1648
Team 1's system 3	0.1975	0.1995	0.1646
Team 1's system 4	0.1904	0.1844	0.1652
Team 1's system 5	0.1192	0.1556	0.1447
Team 1's system 6	0.1836	0.1956	0.1614
Team 1's system 7	0.1806	0.1943	0.1606
Team 1's system 8	0.1549	0.1927	0.1581
Team 1's system 9	0.1857	0.2003	0.1696
Team 1's system 10	0.2052	0.1989	0.1579
Team 2's system 1	0.3990	0.3879	0.2830
Team 2's system 2	0.2186	0.2189	0.1435
Team 2's system 3	0.3689	0.3429	0.1110

Methods 1 through 3 obtained higher scores than the original knn method. This indicates that the modifications in these methods were effective.

Baselines 1 and 2 obtained very low f-measures. This means that the problems handled in this paper were very difficult. Therefore, while the f-measure of the best method 1 (0.4420) was not that high, it was very high when compared with the scores of baselines 1 and 2.

- When we compared the similarity calculation methods, BM25 had the best score. The difference between BM25 and the other method was large (greater than 0.04). We confirmed that BM25 was more effective than the other methods using a statistical test at a significance level of 0.01.

7.2 Experiment in the F-term categorization subtask of the NTCIR Patent Workshop

In this section, we describe the results of our participation in the NTCIR Patent Workshop [3].

The results of all the teams are shown in Table 3. Our team used method 1, BM25, and the abstract, claim, domain and solution regions.[2] Three teams, including ours, participated in the workshop. As shown in Table 3, we obtained the best scores, and the differences between our scores and the other teams' scores were large. This indicates that our methods were effective.[3]

7.3 Experiments in the theme categorization subtask

We next performed the experiments in the theme categorization subtask. We used variations of the k-nearest neighborhood method and various methods of calculating similarity. We extracted terms from the region of the abstract.

The results are shown in Table 4. We tested various values for each parameter and experimented using all combinations of these values to determine the best values. The evaluation scores and parameter values that resulted in the highest scores for each method during the dry run data are shown in Table 4.

We also experimented on the formal run data using the parameters values that resulted in the highest scores for each method during the dry run data. The results are also shown in Table 5.

We used the dry run data to determine the parameters, and the formal run data to evaluate the effectiveness of the methods.

[2] The five systems used in the NTCIR Patent Workshop each had different parameters as shown in the following table.

System	Parameters
Our system 1	$k = 501, k_r = 0.99, k_p = 0.3$
Our system 2	$k = 501, k_r = 1, k_p = 0.3$
Our system 3	$k = 501, k_r = 0.99, k_p = 0.2$
Our system 4	$k = 501, k_r = 0.99, k_p = 0.4$
Our system 5	$k = 501, k_r = 0.95, k_p = 0.3$

The systems are not exactly equal to the system used in Section 7.1 in a detailed architecture. Their evaluation scores are also different, even when they use the same parameters.

[3] The detail of the methods used by the other teams will be published in the proceeding of the NTCIR-5 Patent Workshop [7].

Table 4. Results of the theme categorization dry run

Similarity method	Parameters	A-Precision	R-Precision	F-measure
Baseline 1				
		0.0029^{--}	0.0016^{--}	—-
Baseline 2				
		0.0328^{--}	0.0151^{--}	—-
kNN				
BM25	$k = 5$	0.5697^{--}	0.5320^{--}	0.3691^{--}
overlap	$k = 5$	0.4261^{--}	0.3818^{--}	0.2220^{--}
Method 1				
BM25	$k = 101, k_r = 0.9, k_p = 0.5$	0.6641*	0.5898*	0.5657*
overlap	$k = 101, k_r = 0.99, k_p = 0.6$	0.5713^{--}	0.4869^{--}	0.4729^{--}
Method 2				
BM25	$k = 13, k_u = 0.2$	0.6272^{--}	0.5640^{--}	0.5164^{--}
overlap	$k = 15, k_u = 0.2$	0.5079^{--}	0.4476^{--}	0.3995^{--}
Method 3				
BM25	$k = 7$	0.5974^{--}	0.5451^{--}	0.4675^{--}
overlap	$k = 7$	0.4508^{--}	0.4062^{--}	0.3497^{--}

Table 5. Results of the theme categorization formal run

Similarity method	Parameters	A-Precision	R-Precision	F-measure
Baseline 1				
		0.0015^{--}	0.0005^{--}	—-
Baseline 2				
		0.0251^{--}	0.0191^{--}	—-
kNN				
BM25	$k = 5$	0.5492^{--}	0.5046^{--}	0.3993^{--}
overlap	$k = 5$	0.4192^{--}	0.3716^{--}	0.2377^{--}
Method 1				
BM25	$k = 101, k_r = 0.9, k_p = 0.5$	0.6427*	0.5649*	0.5410*
overlap	$k = 101, k_r = 0.99, k_p = 0.6$	0.5517^{--}	0.4658^{--}	0.4482^{--}
Method 2				
BM25	$k = 13, k_u = 0.2$	0.6127^{--}	0.5542^{--}	0.5086^{--}
overlap	$k = 15, k_u = 0.2$	0.4930^{--}	0.4279^{--}	0.3898^{--}
Method 3				
BM25	$k = 7$	0.5806^{--}	0.5340^{--}	0.4757^{--}
overlap	$k = 7$	0.4473^{--}	0.3926^{--}	0.3460^{--}

We used the two-sided t-test to determine significant differences, with the best methods (method 1 and BM25) as the baseline methods. The baseline methods were labeled "$*$". When a method performed better than the baseline method at a significance level of 0.05 or 0.01, it was labeled "$+$" or "$++$". Likewise, when a method performed worse than the baseline method at a significance level of 0.05 or 0.01, it was labeled "$-$" or "$--$".

Table 5 shows the following.

- When we compared the variations in the k-nearest neighbor method, method 1 had the best score. The difference between method 1 and the other three methods was very small (about 0.01). However, we confirmed that it was more effective than the other methods using a statistical test, at a significance level of 0.01.

  Methods 1 through 3 obtained higher scores than the original knn method. This indicates that the modifications in these methods were effective.

  Baselines 1 and 2 obtained very low f-measures. This means that the problems handled in this paper were very difficult. Therefore, even though the f-measure of the best method 1 (0.5410) was not that high, it was very high when compared with the baselines 1 and 2 scores.

- When we compared the similarity calculation method, BM25 had the best score. The difference between BM25 and the other method was large (greater than 0.04). We confirmed that BM25 was more effective than the other methods using a statistical test at a significance level of 0.01.

In the theme categorization subtask, the best scores were 0.6872, 0.5943, and 0.5269 for average precision, r-precision, and F-measures, respectively. Our best score for the F-measure was 0.5410, thus indicating that our system could obtain a higher F-measure than the best system. Our best average precision and r-precision scores were 0.6427 and 0.5649, respectively, indicating that our system could not obtain a higher average precision and r-precision than the best system. However, our system obtained a relatively high average precision and r-precision.

8 Conclusion

Patent processing is important for fields such as industry, business, and law. We participated in the F-term categorization subtask during NTCIR-5. During this subtask, we classified patent documents into their F-terms using the k-nearest neighbor method. The F-term categories are both extremely precise and useful for classifying patent documents. We used five systems in the F-term categorization subtask. They obtained the best f-measures of all the 18 systems entered. This indicates that our method was effective. After the contest, we performed the experiments in the theme categorization subtask, although we did not officially participate in it. The results showed that our system obtained higher f-measures than the highest f-measures in the systems used in the subtask. This also indicates that our method was effective.

In the future, we would like to construct application systems that could show users the results of classifying patent documents by applying the automatic F-term classification technique used in this study.

References

[1] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[2] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press Inc., 1972.

[3] M. Iwayama, A. Fujii, and N. Kando. Overview of classification subtask at ntcir-5 patent retrieval task. *Proceedings of the Fifth NTCIR Workshop*, 2005.

[4] JPO. Japan patent office. 2005. www.jpo.go.jp/ index.html.

[5] D. D. Lewis. An evaluation of phrasal and clustered representations on a tex categorization task. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1992.

[6] M. Murata, K. Uchimoto, H. Ozaku, Q. Ma, M. Utiyama, and H. Isahara. Japanese probabilistic information retrieval using location and category information. *The Fifth International Workshop on Information Retrieval with Asian Languages*, pages 81–88, 2000.

[7] NTCIR committee. the patent task in the ntcir-5 workshop. 2005.

[8] E. S. Ristad. Maximum Entropy Modeling for Natural Language. ACL/EACL Tutorial Program, Madrid, 1997.

[9] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994.

[10] M. Utiyama. Information retrieval module for ruby, 2005. www2.nict.go.jp/jt/a132/members/mutiyama/software.

[11] M. Utiyama and H. Isahara. Large scale text classification. *9th Annual Meeting of the Association for Natural Language Processing*, 2003. (in Japanese).

[12] Y. Yang and X. Liu. A re-examination of text categorization methods. In *Proceedings of the 22th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999.