# A Passage Retrieval System using Query Expansion and Emphasis

Hiroki Tanioka Kenichi Yamamoto
JustSystems Corporation
Brains Park Tokushima-shi, Tokushima 771-0189, Japan
{hiroki_tanioka, kenichi_yamamoto}@justsystem.co.jp

## Abstract

*We developed a patent retrieval system with the corresponding very large number of patents from NTCIR-6 Patent Retrieval Task. And we developed a method of refining and emphasizing query. Our retrieval system consisting of four PCs could make indices of all claims in specifications for ten years. Then we confirmed that the query emphasis was better mean average precision than merely query expansion. And we had tried to reduce the number of results with the belief assessment.*
**Keywords:** *query expansion, support vector machines, vector space model, inverted file*

## 1 Introduction

Our purposes to participate in the NTCIR-6 Patent Retrieval Task [5] are as follows.

- Research and develop an effectiveness of an application of a *query extraction* method for "invalidity search".
- Research and develop an effectiveness of an application of a *query emphasis* method for "invalidity search".
- Research and develop an effectiveness of an application of *query expansion* method for "invalidity search".

A background of the first purpose is that terms are extracted from a claim automatically, because it is high cost to make a query from a claim manually for an invalidity patent search. However there was not a better precision in an experimental confirmation [10].

The second purpose is that a query is weighted by using extracted terms automatically for an invalidity patent search. To confirm that extracted terms can really enhance a precision for an invalidity patent search, this article compares a precision of a query extraction and a query emphasis.

The third purpose is that a query is expanded by using related terms from a whole claims. We should make certain how a query expansion works for an invalidity patent search.

The rest of the article is divided into three sections. In section 2, we describe an architecture of our retrieval model. In section 3, we describe results of formal runs. In section 4, we discuss about results and future works.

## 2 System Description

In this section we describe the architecture of our retrieval model for all the claims of the Japanese patent. Also the system architecture and the model for the Japanese Retrieval Subtask is explained.

### 2.1 Overview

The search system based on the Vector Space Model using an inverted file-based system is developed [1, 13, 6]. It is different from others that the system employs a passage retrieval system in order to a patent retrieval system.

### 2.2 Passage Retrieval System

Due to our system uses a passage retrieval system, we must solve a problem of spatial scalability depend on the number of documents. Our system already had solved in NTCIR-5 by distributing inverted files [10].

The patent retrieval system needs a score of a patent instead of a claim. For this purpose the system selects related claims and calculates a score of a patent containing related claims. There are several methods of selecting and calculating scores: a total score of claims in a patent, an average score of claims in a patent, and a first level score of claims in a patent. Here, the system uses only the first level score in a patent, based on the past experience [4]. We assume that claims are the same as sub-documents.

### 2.3 Retrieval Model

The retrieval model is designed based on the Vector Space Model. And the calculating formula of a score is

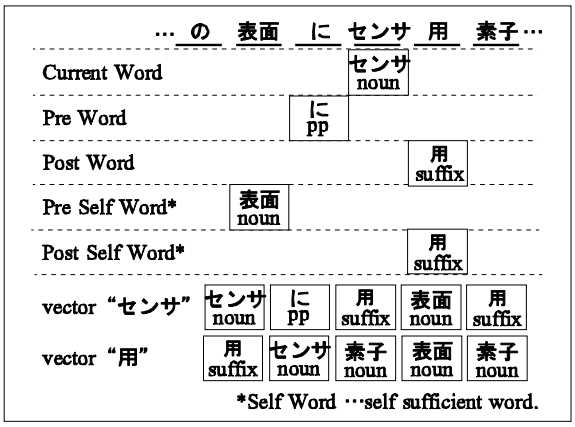


**Figure 1. Features of a SVM for query extraction.**

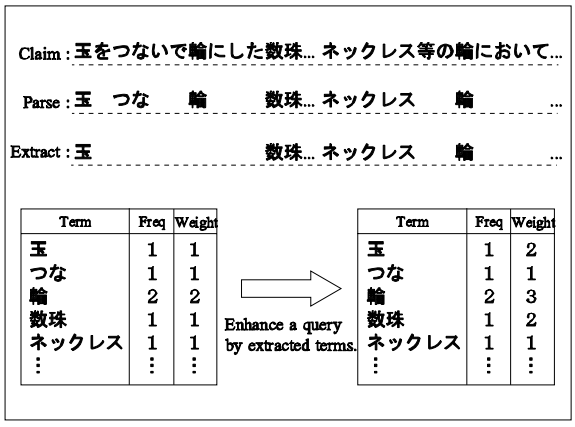


**Figure 2. Query emphasis using query extraction.**

based on a simple calculation of $TF \cdot IDF$ weighting as follows,

$$S_T = (1 + \log(TF)) \cdot \log(\frac{N}{DF}) \quad (1)$$

where $S_T$ is the score of a term $t$, $TF$ is the term frequency, $N$ is the number of documents, and $DF$ is the document frequency. To set the limits, $L_{TF}$ is used instead of $\log(TF)$, when $\log(TF)$ is greater than $L_{TF}$. And $L_{DF}$ is used instead of $log(DF)$, when $log(DF)$ is greater than $L_{DF}$. Where each constant number is declared as follows: $L_{TF} = 7$ and $L_{DF} = 15$.

There are three differences from an original $TF \cdot IDF$ calculating formula.

- Using a logarithm of the term frequency
- Limiting the logarithmic term frequency
- Limiting the logarithmic document frequency

The first difference is based on our pilot study, which shows an adverse effect from a greater values of $TF$ on a $TF \cdot IDF$ calculation. The second difference is a solution of eliminating high frequency terms. And the third difference is a solution of eliminating counts of document which are occurred the term.

The feature of the Vector Space Model contains terms of noun, verb and unknown word as part of speech from all claims by using a tool of a morphological analyzer.[1]

The term weighting is also important. Therefore the *query extraction*, the *query emphasis* and the *query expansion* are used as the term weighting.

[1]The tool of a morphological analyzer using the Hidden Markov Model and the bigram.

## 2.4 Query Extraction

It is the high cost to make a query from a claim manually for the invalidity patent search. In this instance, a query is made from a claim automatically by using a Support Vector Machine (SVM) [11, 7] to choose terms from a claim as a relevant query. For machine learning, LIBSVM [2] is used as a library for SVMs.

The training data of 40 queries are prepared in cooperation with the patent administrator in JustSystems Corporation. The features for a SVM are the surface and the part of speech as follows.

- The word of current (Current Word)
- The word of previous (Pre Word)
- The word of next (Post Word)
- The self-sufficient word of previous (Pre Self Word)
- The self-sufficient word of next (Post Self Word)

And, Figure 1 shows an example of features. Here, “self-sufficient word” is subequal to “content word” in English.

## 2.5 Query Emphasis

The query extraction reduces the terms in the query, and decreases the information content for the information retrieval. However the query emphasis weights the relevant terms in the query, and can expect not to decrease the information content.

In order to emphasize the relevant terms, the query processing appends the relevant terms to the originally parsed terms as shown in Figure 2. This figure shows parsed terms, extracted terms, and a emphasizing process.

| Term | Expanded Terms / Conditional Probability |
|---|---|
| 耕盤 | 苗植/0.132,田/0.068,湿田/0.038,耕起/0.037,深浅/0.034,表土/0.033… |
| ⋮ | |
| パンズ | ハンバーガー/0.523,食パン/0.153,パティ/0.144,バター/0.142,… |
| ｲｿﾏﾙﾃﾞｷｽﾄﾘﾝ | イソマルトース/0.583,ﾏﾙﾄﾃﾞｷｽﾄﾘﾝ/0.388,単糖/0.388,ガラクトース/0.361… |
| イソマルト | キシリトール/0.243,ラクチトール/0.177,マルトース/0.142,ｴﾘｽﾘﾄｰﾙ/0.121,… |
| 蒸し | 茹で/0.054,蒸し器/0.051,蒸煮/0.048,まんじゅう/0.035,中華/0.034… |
| オリゴトース | マルトース/0.376,ﾏﾙﾄﾄﾘｵｰｽ/0.246,糖質/0.202,林原/0.159,… |
| 肉まん | あんまん/0.255,蒸し/0.228,シューマイ/0.227,餃子/0.130,… |
| 林原 | 商事/0.385,プルラン/0.282,トレハオース/0.157,マルトース/0.115,… |
| クロワッサン | デニッシュ/0.326,ペーストリー/0.261,食パン/0.235,バター/0.192,… |
| ピーオー | 東和/0.933,アミコール/0.566,澱/0.566,ＬＬＤ/0.566,名糖/0.566,… |
| デニッシュ | ペーストリー/0.548,クロワッサン/0.389,パイ/0.257,バター/0.248,… |
| 産地 | 中国/0.036,原産/0.035,産出/0.034,生鮮/0.032,直送/0.024,… |
| 面接 | 求人/0.047,求職/0.042,セミ/0.033,人材/0.020,応募/0.020 |
| クッキー | ビスケット/0.146,バター/0.108,チョコレート/0.105,キャンデー/0.072,… |
| ビスケット | クッキー/0.257,チョコレート/0.212,キャンデー/0.115,バター/0.112,… |
| 糖質 | マルトース/0.053,単糖/0.036,ガラクトース/0.026,果糖/0.023,… |
| ラード | 菜種油/0.259,牛脂/0.209,魚油/0.181,綿実油/0.179,バター/0.125, |
| イースト | ペプトン/0.085,製パン/0.081,粉乳/0.076,ショートニング/0.072,… |
| 氷点 | 氷結/0.065,不凍/0.038,凍/0.025,シャーベット/0.015,貯氷/0.014,… |
| ライ麦 | 大麦/0.540,麦/0.239,イネ/0.192,穀類/0.175,エンバク/0.168,… |
| 成虫 | 幼虫/0.292,産卵/0.106,ふ化/0.098,羽化/0.089,死亡/0.064,… |
| ⋮ | |

**Figure 3. Expanded terms using the co-occurrence in claims.**

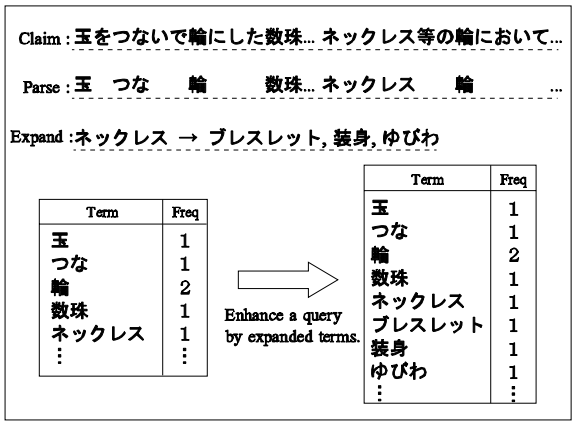


**Figure 4. Query expansion using expanded terms.**

### 2.6 Query Expansion

For the query expansion, a degree of association between a term A and B, is given various ways in earlier studies [3]. Terms are thinned out on the basis of $IDF$ ($> 0.00001$), that means an occurrence rate is no less than 5% in the number of patents.

For the sake of simplicity, The following equation is a simply similarity using a conditional probability:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{DF_{A \cap B}}{DF_B} \qquad (2)$$

where $P(A|B)$ is a conditional probability of a term $A$, given a occurrence of a term $B$ in a same claim, and $DF_{A \cap B}$ is a document frequency as a co-occurrence between $A$ and $B$, $DF_B$ is a document frequency as a occurrence $B$. Hence the probability of $A$ given $B$, is that the number of the co-occurrence between $A$ and $B$ divided by the number of the occurrence $B$.

Figure 3 shows a part of expanded terms, and, Figure 4 shows an example of the query expansion. In the query expansion, expanded terms are reduced by a lower limit of the probability $P(A|B)$ ($\geq 0.1$) and a upper limit of the number of terms ($\leq 5$).

### 2.7 Belief Assessment

To reduce the number of results while keeping the precision, A way of the belief assessment in the post-processing is proposed. First, we make two hypotheses. A result including correct invalidity patents which has a characteristic distribution of $TF \cdot IDF$ score. And we also assume a generalized linear model depend on the score of the first retrieved patent. $f_i$ is a criterion function,

$$f_i = \frac{x_i}{x_0} - t, \quad i \in [0, N), \quad x_0 \neq 0 \qquad (3)$$

where $x_i$ denotes a $TF \cdot IDF$ score at $i$-th in the order of the $TF \cdot IDF$ score, and $t$ is a threshold. If $x_0 is 0$, $f_i is also 0$. (i.e. no retrieved results for a topic, no need to reduce the results.)

$$f_i = \begin{cases} > 0 & \text{if i-th patent has possible} \\ \leq 0 & \text{otherwise} \end{cases}$$

The above equation means that the $i$-th retrieved patent might be an invalidity patent when $f_i$ is greater than $0$. Ten topics in the Japanese Retrieval Subtask were picked beforehand at random to define the threshold $t$. Thereby we assume the threshold $t$ is at least 0.5 for our system, to include correct invalidity patents in a retrieved result.

## 3 Results

In this section, the results of the NTCIR-6 Patent Retrieval Task are shown. Then, this article shows some results of the Japanese Retrieval Subtask and the English Retrieval Subtask.

And now, the runs for the English Retrieval Subtask are almost the same conditions as the Japanese Retrieval Subtask, because of the shortness of time of tuning. Also, any run for the English Retrieval Subtask don't use the query extraction, the query emphasis, and the query expansion.

**Table 1. Specification of PCs**

| | CPU [GHz] | Ram [GB] | OS |
|---|---|---|---|
| A | Celeron 2.4 | 2 | WinXP Pro SP2 |
| B | Celeron 2.4 | 1 | FedoraCore4 |
| C | Celeron 2.4 | 1 | FedoraCore4 |
| D | Celeron 2.2 | 1 | FedoraCore4 |

∗The system is written in Java (JDK1.4).

**Table 2. Differences of each run (1)**

| Run ID | Query Processing and Post-processing |
|---|---|
| JSPAT1 | All terms of query |
| JSPAT2 | Expansion of query |
| JSPAT3 | Extraction using SVM |
| JSPAT4 | Emphasis using SVM |
| JSPAT5 | All terms of query, reduced results |
| JSPAT6 | Expansion of query, reduced results |
| JSPAT7 | Extraction using SVM, reduced results |
| JSPAT8 | Emphasis using SVM, reduced results |

∗JSPAT1˜4 are without the post-processing of reducing results. And JSPAT5˜8 are with the post-processing of reducing results.

**Table 3. Results of each run (1)**

| Run ID | $ALL_A$ | $NTC4_A$ | $NTC5_A$ | $NTC6_A$ |
|---|---|---|---|---|
| JSPAT1 | 4.79 | 5.02 | 6.62 | 3.73 |
| JSPAT2 | 4.81 | 4.92 | 6.63 | 3.75 |
| JSPAT3 | 4.36 | **8.32** | 6.36 | 3.25 |
| JSPAT4 | **4.89** | 6.56 | **6.96** | **3.78** |
| JSPAT5 | 4.76 | 4.23 | 6.59 | 3.71 |
| JSPAT6 | 4.78 | 4.16 | 6.60 | 3.73 |
| JSPAT7 | 4.34 | **8.22** | 6.35 | 3.23 |
| JSPAT8 | **4.87** | 5.71 | **6.95** | **3.76** |

∗$ALL_A$ contains all of topics, $NTC4_A$ is topics at the NTCIR-4, $NTC5_A$ is topics at the NTCIR-5, and $NTC6_A$ is topics at the NTCIR-6.

### 3.1 Japanese Retrieval Subtask

Table 1 shows a specification of these PCs for the Japanese Retrieval Subtask. Also supplementary information is that each network-linked PC connect on gigabit Ethernet over TCP/IP. The system retrieved the candidates of invalidity patents for 3,257 topics in about 20 hours. Hence the average retrieving time was about 22 seconds per a topic.

And, Table 2 shows the differences of eight runs that the JSPAT team submitted. The results from JSPAT1 to JSPAT4 are without the post-processing. The results from JSPAT5 to JSPAT8 are with the post-processing of reducing results. Each result has their own characteristics of the query processing strategy and the post-processing.

On that basis, Table 3 shows mean average precisions (MAPs) of all runs. The raw means respectively, $ALL_A$ is a MAP for all of topics, $NTC4_A$ is a MAP for 34 topics at the NTCIR-4, $NTC5_A$ is a MAP for 1189 topics at the NTCIR-5, and $NTC6_A$ is a MAP for 1685 topics at the NTCIR-6. According to results, the result of JSPAT4 was relatively better than other runs from JSPAT1 to JSPAT4 (Group A), and the result of JSPAT8 was relatively better than other runs from JSPAT5 to JSPAT8 (Group B).

To simply look at the result, The query emphasis is more useful than other query processing strategies for an invalidity patent search. The query emphasis using extracted terms leads to a positive outcome. However the query extraction really enhanced the precision of $NTC4_A$ only.

Otherwise the reduced results were verified by comparing the precisions of two groups (Group A and B). As a result, the precisions of two groups were pretty close, in spite of Group B reduced the considerable number of retrieved patents. In case that JSPAT1 and JSPAT5, the number of results is down by 17.8% (3,253,327 → 2,675,384).

### 3.2 English Retrieval Subtask

The specification is the same the for English Retrieval Subtask shown in Table 1. The system retrieved the candidates of invalidity patents for 2,221 topics in about 9.5 hours. Hence the average retrieving time was about 15.4 seconds per a topic.

And Table 4 shows the differences of four runs that the JSPAT team submitted. The results of JSPAT1 and JSPAT2 are without the post-processing. The results of JSPAT3 and JSPAT4 are with the post-processing of reducing results. Each result has their own characteristics post-processing. And JSPAT2 and JSPAT4 use the application date, i.e. APP-DATE, in order to reduce published patents before the application date.

Although, Table 5 shows that all MAP results were not good. In all probability, a reason of low precisions is attributed mainly to the tuning of parameters and the nonuse of stop words.

### 3.3 Effectiveness of Belief Assessment

The result shows an effectiveness of the belief assessment method in the Japanese Retrieval Subtask and the English Retrieval Subtask. The method just slightly declined about a macro average of declining rate of a MAP. (e.g. The Japanese Retrieval Subtask is 0.032, the English Retrieval Subtask is 0.079, and a total is 0.041.)

The method reduced the number of retrieved results 17.8% in the Japanese Retrieval Subtask and 0.02% in the English Retrieval Subtask. And the number of retrieved results in English Retrieval Subtask could barely reduced. The reason simply comes from the fact that the threshold of the belief assessment method was the same as the Japanese Retrieval Subtask, Hence these results should be excepted from consideration regarding the belief assessment method.

**Table 4. Differences of each run (2)**

| Run ID | Query Processing and Post-processing |
|---|---|
| JSPAT1 | CLAIM(All terms of query) |
| JSPAT2 | CLAIM & APP-DATE |
| JSPAT3 | CLAIM, reduced results |
| JSPAT4 | CLAIM & APP-DATE, reduced results |

∗JSPAT1 and JSPAT2 are without the post-processing of reducing results. JSPAT3 and JSPAT4 are with the post-processing of reducing results. CLAIM means the run uses all terms of the CLAIM tag in a query, and APP-DATE means the runs use the APP-DATE tag about the application date in a query.

**Table 5. Results of each run (2)**

| Run ID | $MAP_a$ |
|---|---|
| JSPAT1 | 1.27 |
| JSPAT2 | 1.26 |
| JSPAT3 | 1.23 |
| JSPAT4 | 1.22 |

∗The best score of MAP in English Retrieval Subtask is 4.17 by AFLAB2, and average score of MAP in English Retrieval Subtask is 2.59. Hence, the results from JSPAT1 to JSPAT4 are really bad results.

## 4 Conclusions

The results of three query processing methods are evaluated and showed some effects in the Japanese Retrieval Subtask. Concretely, the query emphasis using extracted terms slightly enhanced the precision. But in the case of $NTC4_A$, the query extraction using a SVM was the best. It caused that the training data included most claims in $NTC4_A$, and there might be a trend difference between the topics.

The belief assessment really reduced the number of retrieved results (17.8%) without a significant decline of the precision. Hence the belief assessment probably provided a user with comfort. The result remains an issue about the distribution of the correct invalidity patents in retrieved results.

## Acknowledgement

We thank our colleagues for their encouragement. And we appreciate many collaborators, and workshop organizers provided us with work in the area.

## References

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*, chapter 5-9. Addison-Wesley, 1999.

[2] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[3] I. Dagan, L. Lee, and F. C. N. Pereira. Similarity-based models of word cooccurrence probabilities. *Mach. Learn.*, 34(1-3):43–69, 1999.

[4] D. A. Evans and R. G. Lefferts. Design and evaluation of the clarit-trec-2 system. In *TREC*, pages 137–150, 1993.

[5] A. Fujii, M. Iwayama, and N. Kando. Overview of the patent retrieval task at the ntcir-6 workshop. In *Proceedings of the Sixth NTCIR Workshop Meeting*, 2006.

[6] C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

[7] J.-T. N.Cristianini. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[8] D. G. S. Richard O. Duda, Peter E. Hart. *Pattern Classification Second Edition*, chapter 5.11. John Wiley & Sons Inc., 2000.

[9] Y. C. S. Salton G., Wong A. A vector space model for automatic indexing. *Communications of the ACM*, 613-620(18), 1975.

[10] H. Tanioka and K. Yamamoto. A distributed retrieval system for ntcir-5 patent retrieval task. In *Proceedings of the Fifth NTCIR Workshop Meeting*, 2005.

[11] V.N.Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.

[12] J. Wu, H. Tanioka, S. Wang, D. Pan, K. Yamamoto, and Z. Wang. An improved vsm based information retrieval system and fuzzy query expansion. In *FSKD (1)*, pages 537–546, 2005.

[13] J. Zobel and A. Moffat. Inverted files for text search engines. *ACM Comput. Surv.*, 38(2):6, 2006.