A Kernel for Interactive Document Retrieval Based on Support Vector Machines

Hiroshi Murata* †, Takashi Onoda* and Seiji Yamada† ‡
* Central Research Institute of Electric Power Industry
2–11–1 Iwado kita, Komae-shi, Tokyo 201–8511 Japan
Email: {murata,onoda}@criepi.denken.or.jp
† The Graduate University for Advanced Studies (SOKENDAI)
2–1–2 Hitotsubashi, Chiyoda-ku, Tokyo 101–8430 Japan
Email: murata@nii.ac.jp
‡ National Institute of Informatics
2–1–2 Hitotsubashi, Chiyoda-ku, Tokyo 101–8430 Japan
Email: seiji@nii.ac.jp

***Abstract*—This paper describes an application of support vector machines (SVMs) to interactive document retrieval using active learning. We show that an SVM-based retrieval has an association with conventional Rocchio-based relevance feedback by a comparative analysis. We propose a cosine kernel, which denotes cosine similarity, suitable for an SVM-based interactive document retrieval based on the analysis. We confirm the effectiveness of our approach when we adopt Boolean, TF, and TFIDF to represent the document vectors. Our proposed approach, in particular, TF representation, shows better performance, which is demonstrated experimentally.**

I. INTRODUCTION

With the development of information technology, the amount of text data is increasing explosively and document retrieval is expected to become more sophisticated. The task of finding several relevant documents is known as document retrieval. It can also be defined as a task to find as many relevant documents as possible, even if the system strains a user. This latter task is the focus of our study. Document retrieval systems, which use information on interactive user feedback, have been studied in many ways [3] [4].

In most frameworks for information retrieval, a vector space model (VSM) is used in which a document is described using a high-dimensional vector [10]. An information retrieval system using a VSM computes the degree of relevance between a query vector and document vectors by using the cosine similarity of the two vectors, and then provides a list of retrieved documents to a user.

In general, since a user rarely describes a query precisely in the first attempt, an interactive approach has been proposed to modify a query vector on the basis of a user's evaluation of documents in a list of retrieved documents. This method is called *relevance feedback* [9] and is used widely in information retrieval systems. In this method, a user directly evaluates whether a document in a list of retrieved documents is relevant or irrelevant, and the system modifies the query vector on the basis of the user's evaluation. A conventional way to modify a query vector is through a simple learning rule that reduces the difference between the query vector and the documents evaluated as relevant by a user.

A method of relevance feedback based on VSM is the Rocchio algorithm [9]. A new query vector Q_{m+1} is calculated using the following equation:

$$Q_{m+1} = Q_m + \beta \sum_{\mathbf{x} \in R_r^m} \mathbf{x} - \gamma \sum_{\mathbf{x} \in R_n^m} \mathbf{x}. \qquad (1)$$

Here, $\mathbf{x}$ represents document vectors, and R_r^m and R_n^m are document sets that are determined as relevant and irrelevant when feedback is obtained m times. β and γ are parameters that adjust the relative impact of relevance and irrelevance. The Rocchio algorithm evaluates the degree of relevance using the cosine similarity between a new query vector Q_{m+1} and document vectors.

Another approach has been proposed in which relevant and irrelevant document vectors are classified as positive and negative examples for a target concept based on classification learning [7]. Some studies have proposed that support vector machines (SVMs) [13] with excellent ability to classify examples into two classes can be applied to classification learning of relevance feedback [2].

We have proposed a relevance feedback framework with an SVM for active learning and experimentally demonstrated the usefulness of our proposed method [8]. In contrast to a conventional relevance feedback system which shows a list of the most relevant documents to a user, our system provides a list of the most relevant documents that are difficult for the SVM to classify. In such a system, the degree of relevance is evaluated using a signed distance from the optimal hyperplane.

It is not clear how the signed distance in the SVM has characteristics of the VSM. Hence, we formulate the degree of relevance using signed distance in the SVM and make a comparative analysis with the Rocchio algorithm. We propose a cosine kernel which expresses distance in the SVM using cosine similarity.

In the remainder of this paper, we describe an information retrieval system using an SVM-based relevance feedback in

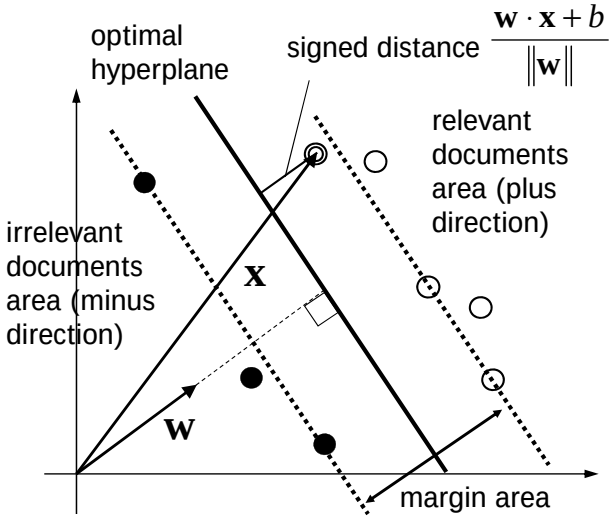

Fig. 1. Relevance feedback based on SVM

section II and propose the cosine kernel in section III. To evaluate the effectiveness of our approach, we present the result of experiments performed using a TREC data set in section IV. Finally, we introduce related work in section V and conclude our work in section VI.

II. SVM-BASED INTERACTIVE DOCUMENT RETRIEVAL

A. SVM-based relevance feedback

Figure 1 shows the concept of relevance feedback based on an SVM. In Figure 1, white and black circles represent documents that have been decided as relevant and irrelevant, respectively. Let us consider decided document vectors $\mathbf{x}_i$ having class labels $y_i = \pm 1$; a linear classifier for an undecided document vector $\mathbf{x}$ is of the form

$$f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + b. \tag{2}$$

The signed distance between the decided documents and hyperplane (margin) is

$$\min_i \frac{\mathbf{w}\cdot\mathbf{x}_i + b}{||\mathbf{w}||}. \tag{3}$$

The minimum distance between decided documents and hyperplane is

$$\min_i \frac{|\mathbf{w}\cdot\mathbf{x}_i + b|}{||\mathbf{w}||}. \tag{4}$$

Here, we add the constraint $\min_i |\mathbf{w}\cdot\mathbf{x}_i + b| = 1$; thus, Equation (4) can be rewritten as $1/||\mathbf{w}||$.

The hyperplane having maximum margin can be found by solving the following quadratic programming problem $\tau(\mathbf{w}) = ||\mathbf{w}||^2$, which is subject to the inequality constraints $y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 (i = 1, \ldots, \ell)$.

We construct a Lagrangian for solving the above equation as follows:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{\ell} \alpha_i (y_i((\mathbf{x}_i\cdot\mathbf{w}) + b) - 1) \tag{5}$$

where $\alpha_i \geq 0$ represents Lagrange multipliers. The minimization over $\mathbf{w}$ and b can be achieved by the following differentiation:

$$\frac{\partial L}{\partial b} = 0 \quad \frac{\partial L}{\partial \mathbf{w}} = 0.$$

Therefore we have the conditions

$$\sum_{i=1}^{\ell} \alpha_i y_i = 0 \tag{6}$$

$$\mathbf{w} = \sum_{i=1}^{\ell} \alpha_i y_i \mathbf{x}_i. \tag{7}$$

By using the previous results we obtain

$$W(\boldsymbol{\alpha}) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \tag{8}$$

subject to

$$\alpha_i \geq 0 (i = 1, \ldots, \ell), \sum_{i=1}^{\ell} \alpha_i y_i = 0. \tag{9}$$

The documents $\mathbf{x}_i$ at $\alpha_i = 0$ have no influence on optimizing the hyperplane. Only the documents $\mathbf{x}_i$ that are shown by circles on dashed lines in Figure 1 decide the optimal hyperplane. These documents are obtained under the condition $\alpha > 0$. This $\alpha_i > 0$ data are called support vectors. The distance between the document and optimal hyperplane is defined as the degree of relevance in SVM-based relevance feedback.

B. Comparative analysis of relevance feedback

The signed distance between the optimal hyperplane and an undecided document vector, which is shown by a double circle in Figure 1, can be expressed as

$$\begin{aligned} \frac{\mathbf{w}\cdot\mathbf{x} + b}{||\mathbf{w}||} &= \frac{||\mathbf{w}||||\mathbf{x}|| \cos\theta_w + b}{||\mathbf{w}||} \\ &= ||\mathbf{x}|| \cos\theta_w + \frac{b}{||\mathbf{w}||} \end{aligned} \tag{10}$$

where θ_w is the angle between $\mathbf{w}$ and $\mathbf{x}$.

Hence, labels of relevant and irrelevant documents are $y_i = 1$ and $y_i = -1$ respectively,

$$\mathbf{w} = \sum_j \alpha_j \mathbf{x}_j - \sum_k \alpha_k \mathbf{x}_k \tag{11}$$

where j and k are indices of relevant and irrelevant documents respectively.

The query update equation of the Rocchio algorithm, as shown in Equation (1), can be transformed as follows:

$$Q_{m+1} = Q_0 + \sum_j \beta \mathbf{x}_j - \sum_k \gamma \mathbf{x}_k \tag{12}$$

where Q_0 is the initial query vector which is derived from a user's initial input query.

Comparison of Equation (11) and (12) shows that the vector $\mathbf{w}$ equation of SVM-based relevance feedback is equivalent to query update equation of the Rocchio algorithm when the initial query vector is the zero vector.

Initial retrieved documents are determined by

The degree of relevance determines the documents to be retrieved initially. The cosine similarity of the initial query is used in an SVM-based relevance feedback and Rocchio algorithm to determine the degree of relevance. Therefore, initially retrieved documents contain words of the initial query and there is negligible difference between Equation (11) and (12).

Moreover, in equation (11), vector $\mathbf{w}$ only consists of documents of support vectors for which $\alpha_i \neq 0$. However, most documents have a tendency to become support vectors experimentally.

The above discussion suggests that the equation of vector $\mathbf{w}$ for an SVM-based relevance feedback is equivalent to the query update equation of the Rocchio algorithm, and vector $\mathbf{w}$ is similar to the updated query vector of the Rocchio-based relevance feedback.

Furthermore, in Equation (11), SVM-based method decides weights of individual document vectors as α_i. This weight α_i is calculated to minimize structural risk under zero empirical risk. On the other hand, in the Rocchio-based method, the individual document vectors have the same weights as β and γ. These parameters are decided by trial and error.

C. Improvement of document representation based on comparative analysis

In the previous section, we showed that equation $\mathbf{w}$ of relevance feedback based on an SVM is equivalent to the query update equation of the Rocchio algorithm. The degree of relevance in SVM-based method evaluates $||\mathbf{x}|| \cos\theta_w$ from Equation (10) because unique values of $\mathbf{w}$ and b are determined from decided document vectors. On the other hand, the degree of relevance in Rocchio-based method is evaluated as $\cos\theta_q$ which denotes cosine similarity of the angle θ_q between the document vector and query vector.

The comparison suggests that the degree of relevance in the SVM-based method increases as the target document vector $||\mathbf{x}||$ is large. This means that the degree of relevance increases with large $||\mathbf{x}||$ rather than small θ_w. Therefore, a document which has many words has high relevance, and a document which has fewer words has low relevance. To avoid this problem, we define the degree of relevance in the SVM-based method as genuine cosine similarity $\cos\theta_w$. A simple method to realize this is by normalizing a document vector.

III. PROPOSAL OF COSINE KERNEL BASED ON COMPARATIVE ANALYSIS

We substitute a kernel K for the dot product in Equation (8). The kernel corresponds to a dot product in some feature space that is related to the input space via a nonlinear map Φ which can have a high dimensionality,

$$K(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}'). \tag{13}$$

By using K, the classifier takes the form (cf. Equations (2), (7))

$$\begin{aligned} f(\Phi(\mathbf{x})) &= \mathbf{w} \cdot \Phi(\mathbf{x}) + b \\ &= \sum_{i=1}^{\ell} \alpha_i y_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) + b \\ &= \sum_{i=1}^{\ell} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b. \end{aligned} \tag{14}$$

The dual optimization problem of Equation (8) is rewritten to maximize

$$W(\boldsymbol{\alpha}) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j), \tag{15}$$

with the same constraints.

We consider that the kernel $K(\mathbf{x}, \mathbf{x}')$ has cosine similarity when the angle between the two vectors $\mathbf{x}$ and $\mathbf{x}'$ is θ such that

$$K(\mathbf{x}, \mathbf{x}') = \cos\theta = \frac{\mathbf{x} \cdot \mathbf{x}'}{||\mathbf{x}||||\mathbf{x}'||}. \tag{16}$$

It can be seen from this equation that using cosine similarity as a kernel is equivalent to normalizing the document vector. We call this the cosine kernel.

IV. EXPERIMENTS

A. Experimental setting

The document data set we used is a set of articles in an ad hoc task, which was widely used in the 6th, 7th and 8th text retrieval conference (TREC). The data set has about 530,000 newspaper articles. Each TREC provides 50 retrieval problems and information on relevant documents for each retrieval problem. Hereafter, we call the retrieval problem the "topic." In our experiments, 150 topics are tested. Each topic has three tags: a title tag, a description tag, and a narrative tag. The title tag has two or three terms to describe the topic. The description tag introduces the topic. The narrative tag reports the topic. Our experiments used two or three terms of the title tag as a query. Our experiments also removed stopwords and created stemming for documents and queries

We adopted Boolean, TF, and TFIDF methods to represent the document vector. The adopted TFIDF is as follows:

$$w(t,d) = \frac{\log(\mathrm{tf}(t,d)+1)}{\log(\mathrm{uniq}(d))} \log\frac{N}{\mathrm{df}(t)}. \tag{17}$$

The notations used in the above equation are as follows:

- $w(t,d)$ is the weight of a term t in a document d,
- $\mathrm{tf}(t,d)$ is a frequency of a term t in a document d,
- N is the total number of documents in a data set,
- $\mathrm{df}(t)$ is the number of documents including a term t,
- $\mathrm{uniq}(d)$ is the number of different terms in a document d.

We use the selection rule of displayed documents proposed in [8], which are displayed to a user for their classification.

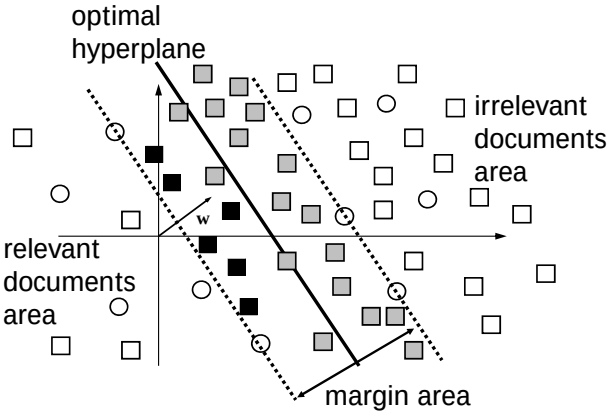

Fig. 2. Proposed selection heuristics

Proposed Selection Rule All documents are mapped onto the feature space. The learned SVM classifies the documents as relevant or irrelevant. The documents that are decided to be relevant and within the margin area of the SVM are selected. The top N ranked documents, which are ranked using the distance from the optimal hyperplane, are displayed to a user as results of the system's information retrieval (see Figure 2 in which circles represent evaluated documents, squares represent non-checked documents, and black squares represent selected documents). This rule is expected to achieve most effective retrieval performance. We propose the use of this rule for retrieval document based on relevance feedback.

The size N of retrieved and displayed documents at each iteration was set as 10 or 20. The feedback iterations m were decided by the total number of displayed documents. In these experiments, we set the total number of displayed documents to 100, which includes initial search documents. When the size N of retrieved and displayed documents at each iteration is 10, the feedback iterations m are between one and nine. When N is 20, the feedback iterations m are between one and four.

To compare the learning performance of our proposed method with other methods, we evaluated the following criterion.

P30: Precision within the top 30 documents, which is the proportion of relevant documents in the top 30 documents [5].

To compare the retrieval performance of our proposed method with the other methods, we evaluated the following criterion.

P: Precision of all displayed documents, where

$$P = \frac{N_{rel}}{N_{dis}}$$

where N_{rel} denotes the number of relevant documents among all displayed documents and N_{dis} denotes the total number of displayed documents.

B. Experimental results

Figures 3 and 4 show the values of P. Figures 5 and 6 show the values of P30.

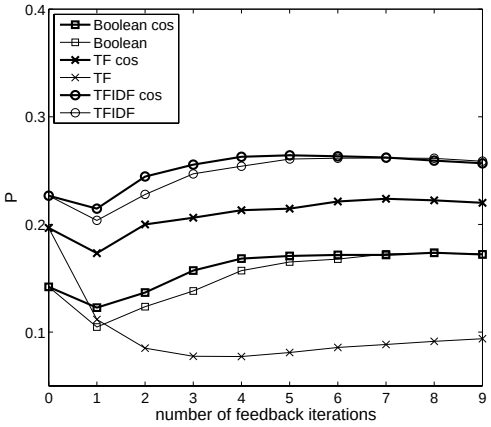

Fig. 3. Results of retrieval performance using criterion P (displayed documents $N = 10$).

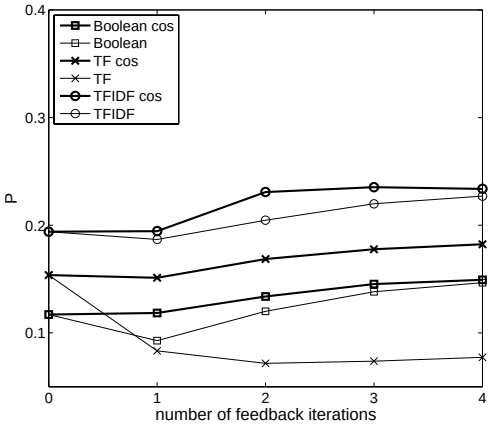

Fig. 4. Results of retrieval performance using criterion P (displayed documents $N = 20$).

TABLE I
COMPARISON OF DISPLAYED VECTOR LENGTHS $||\mathbf{x}||$ USING SVM WHEN $m = 1, N = 10$.

	mean	median	min	max	SD
Boolean	23.0	16.1	2.6	118.7	20.1
TF	248.4	77.7	3.5	12258.6	724.2
TFIDF	29.1	16.2	3.2	228.4	32.9

SD:Standard Deviation

In these figures, zero in a number of feedback iteration implies the performance of the initial retrieval, bold lines indicate the results of the cosine kernel, and solid lines indicate the results of a linear SVM.

These figures suggest that the cosine kernel shows better performance, in particular, when used with the TF method (comparison of lines with crosses in figures).

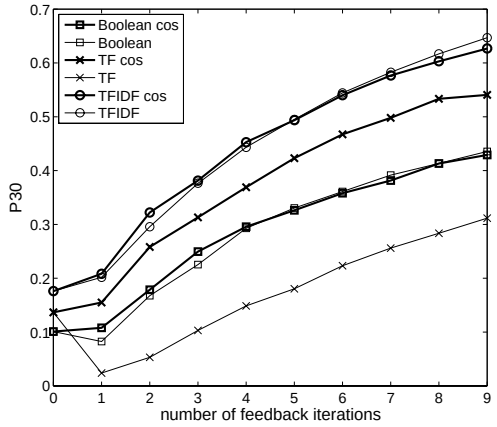

Fig. 5. Results of learning performance using criterion $P30$ (displayed documents $N = 10$).

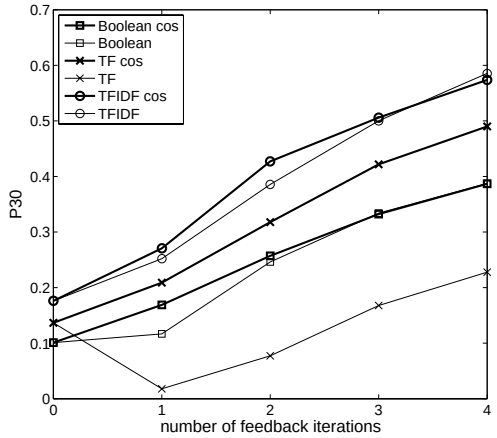

Fig. 6. Results of learning performance using criterion $P30$ (displayed documents $N = 20$).

C. Discussion

The above figures suggest that TF representation helps to achieve a large improvement in document retrieval. TF representation is strongly influenced by the vector length. Table I shows a comparison of displayed vector lengths $||\mathbf{x}||$ using SVM when $m = 1, N = 10$. This table shows that TF representation has some large length vectors compared with other representations. We consider these large vectors to have a negative impact on retrieval accuracy. The cosine kernel evaluates the angle between vectors; hence, the cosine kernel improves retrieval performance.

On the other hand, Boolean and TFIDF approaches gradually becoming reduce the efficiency of results as iteration progresses. For this reason, the classification accuracy of the discriminant hyperplane by an SVM gradually increases due to an iteration feedback process, and the display order is not changed by $||\mathbf{x}||$.

V. RELATED WORK

Several studies have been conducted for an SVM-based relevance feedback. Drucker et al. applied an SVM for relevance feedback and confirmed its effectiveness when there are few relevant documents in the document database[2]. They experimented with differences between document vectors the rate of change of relevant documents in the document database. However, they did not discuss the degree of relevance in detail. We show the effectiveness of the cosine kernel by a comparative analysis with the Rocchio-based method.

Tong and Koller studied active learning for text classification based on an SVM [12]. They proposed a sample selection for effective cutting of the version space based on margin and experimentally showed its effectiveness. However, they discussed only the application of text classification; they did not inspect the method from the viewpoint of interactive document retrieval.

The weights in the Rocchio-based method are decided by trial and error, even for the improved method[1] [11]. We show that the weights based on an SVM are decided by means of similarity between an SVM-based relevance feedback and the Rocchio-based method.

In the study of text classification applying Rocchio-based method, an automatic decision of weights was made by maximizing the break-even point in which recall is equal to the precision[6]. This method needs training documents and evaluation documents; hence, the application of relevance feedback is difficult.

A study was also conducted to decide the weights for relevance feedback [5]. The study proposed a method to decide feedback weight α which decides the balance between queries and feedback documents, in Kullback-Leibler divergence retrieval model by machine learning. In this study, the weight α is decided by logistic regression using features selected by heuristics. However, this method needs training queries, and α changes with the queries. Our proposed method automatically decides weights using feedback results. Therefore, our method is more feasible.

VI. CONCLUSION

In this paper, we propose a cosine kernel which denotes cosine similarity, suitable for an SVM-based interactive document retrieval based on a comparative analysis with a Rocchio-based method.

Our proposed approach, in particular, TF representation, shows better performance, which is demonstrated experimentally.

REFERENCES

[1] C. Buckley and G. Salton: Optimization of relevance feedback weights, Proc. of the 18th annual Int. ACM SIGIR Conf., pp. 351–357 (1995).
[2] H. Drucker, B. Shahrary, and D. C. Gibbon: Support vector machines: relevance feedback and information retrieval, Information Processing and Management, **38**, pp. 305–323 (2002).
[3] P. Ingwersen: Information Retrieval Interaction, Taylor Graham (1992).

[4] J. Koenemann and N. J. Belkin: A case for interaction: a study of interactive information retrieval behavior and effectiveness, Proc. 27th Annual SIGCHI Conf. on Human factors in Computing Systems, pp. 205–212 (1996).
[5] Y. Lv and C. Zhai: Adaptive relevance feedback in information retrieval, Proc. of the 18th ACM CIKM, pp. 255–264 (2009).
[6] A. Moschitti: A study on optimal parameter tuning for Rocchio text classifier, Proc. of the 25th European Conf. on Information Retrieval Research (ECIR03), pp. 420–435 (2003).
[7] M. Okabe and S. Yamada: Learning filtering rulesets for ranking refinement in relevance feedback, Knowledge-Based Systems, **18**, 2-3, pp. 117–124 (2005).
[8] T. Onoda, H. Murata, and S. Yamada: SVM-based interactive document retrieval with active learning, New Generation Computing, **26**, 1, pp. 49–61 (2008).
[9] G. Salton Ed.: Relevance feedback in information retrieval, pp. 313–323, Englewood Cliffs, N.J.: Prentice Hall (1971).
[10] G. Salton and J. McGill: Introduction to modern information retrieval, McGraw-Hill (1983).
[11] A. Singhal, M. Mika, and C. Buckley: Learning routing queries in a query zone, ACM SIGIR Forum, 31, pp. 25–32 (1997).
[12] S. Tong and D. Koller: Support vector machine active learning with applications to text classification, J. of Machine Learning Research, **2**, pp. 45–66 (2001).
[13] V. Vapnik: Statistical learning theory, Wiley, New York (1998).