社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－ＩＣＳ－133 (3)
2003／9／14

Document Retrieval based on Relevance Feedback with Active Learning

TakashiOnoda† HiroshiMurata† SeijiYamada††

† Central Research Institute of Electric Power Industry, Comm. & Info. Lab. 2–11–1, Iwado Kita, Komae-shi, Tokyo, 201–8511 JAPAN
†† National Institute of Informatics 2–1–2, Hitotsubashi, Chiyoda-ku, Tokyo, 101–8430 JAPAN
E-mail: †{onoda,murata}@criepi.denken.or.jp, ††seiji@nii.ac.jp

abstract We investigate the following data mining problems from the document retrieval: From a large data set of documents, we need to find documents that relate to human interest as few iterations of human testing or checking as possible. In each iteration a comparatively small batch of documents is evaluated for relating to the human interest. We apply active learning techniques based on Support Vector Machine for evaluating successive batches, which is called *relevance feedback*. Our proposed approach has been very useful for document retrieval with relevance feedback experimentally. In this paper, we adopt several representations of the Vector Space Model and several selecting rules of displayed documents at each iteration, and then show the comparison results of the effectiveness for the document retrieval in these several situations.
Key word Relevance Feedback, Document Retrieval, Support Vector Machine, Active Learning

能動学習を伴う適合フィードバックに基づく文書検索

小野田 崇†, 村田 博士†, and 山田 誠二††

† (財) 電力中央研究所情報研究所 〒 201–8511 東京都狛江市岩戸北 2–11–1
†† 国立情報学研究所 〒 101–8430 東京都千代田区一ツ橋 2–1–2
E-mail: †{onoda,murata}@criepi.denken.or.jp, ††seiji@nii.ac.jp

あらまし 文書検索の分野では，最近，適合フィードバック文書検索が注目を集めている．そのような状況の中，我々はサポートベクターマシンを用いた適合フィードバック文書検索手法を提案し，その有効性をベンチマークデータを使って実験的に示してきた．本報告では，提案してきた方法におけるベクトル空間モデルの表現方法および，ユーザに提示し，評価してもらう文書選択ルールの違いによる検索効率と学習効率の違いについて述べる．
キーワード 適合フィードバック, 文書検索, サポートベクターマシン, 能動学習

1. Introduction

As progression of the internet technology, accessible information by end users is explosively increasing. In this situation, we can now easily access a huge document database through the WWW. However it is hard for a user to retrieve relevant documents from which he/she can obtain useful information, and a lot of studies have been done in information retrieval, especially document retrieval [1]. Active works for such document retrieval have been reported in TREC(Text Retrieval Conference) [2] for English documents, IREX(Information Retrieval and Extraction Exercise) [3] and NTCIR(NII-NACSIS Test Collection for Information Retrieval System) [4] for Japanese documents.

In most frameworks for information retrieval, a Vector Space Model(which is called VSM) in which a document is described with a high-dimensional vector is used [5]. An information retrieval system using a vector space model computes the similarity between a query vector and document vectors by cosine of the two vectors and indicates a user a list of retrieved documents.

In general, since a user hardly describes a precise query in the first trial, interactive approach to modify the query vector by evaluation of the user on documents in a list of retrieved documents. This method is called *relevance feedback* [6] and used widely in information retrieval systems. In this method, a user directly evaluates whether a document is relevant or irrelevant in a list of retrieved documents, and a

system modifies the query vector using the user evaluation. A traditional way to modify a query vector is a simple learning rule to reduce the difference between the query vector and documents evaluated as relevant by a user.

In another approach, relevant and irrelevant document vectors are considered as positive and negative examples, and relevance feedback is transposed to a binary classification problem [7]. For the binary classification problem, Support Vector Machines(which are called SVMs) have shown the excellent ability. And some studies applied SVM to the text classification problems [8] and the information retrieval problems [9].

Recently, we have proposed a relevance feedback framework with SVM as *active learning* and shown the usefulness of our proposed method experimentally [10]. Now, we are interested in which is the most efficient representation for the document retrieval performance and the learning performance, boolean representation, TF representation or TFIDF representation, and what is the most useful selecting rule for displayed documents at each iteration. In this paper, we adopt several representations of the Vector Space Model and several selecting rules of displayed documents at each iteration, and then show the comparison results of the effectiveness for the document retrieval in these several situations.

In the remaining parts of this paper, we explain a SVM algorithm in the second section briefly. An active learning with SVM for the relevance feedback, and our adopted VSM representations and selecting displayed documents rules are described in the third section. In the fourth section, in order compare the effectiveness of our adopted representations and selecting rules, we show our experiments using a TREC data set of Los Angeles Times and discuss the experimental results. Eventually we conclude our work in the fifth section.

2. Support Vector Machines

Formally, the Support Vector Machine (SVM) [11] like any other classification method aims to estimate a classification function $f : \mathcal{X} \rightarrow \{\pm 1\}$ using labeled training data from $\mathcal{X} \times \{\pm 1\}$. Moreover this function f should even classify unseen examples correctly.

For SV learning machines that implement linear discriminant functions in feature spaces, the capacity limitation corresponds to finding a large margin separation between the two classes. The margin ϱ is the minimal distance of training points $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathbf{R}, y_i \in \{\pm 1\}$ to the separation surface, i.e. $\varrho = \min_{i=1,\ldots,\ell} \rho(\mathbf{z}_i, f)$, where $\mathbf{z}_i = (\mathbf{x}_i, y_i)$ and $\rho(\mathbf{z}_i, f) = y_i f(\mathbf{x}_i)$, and f is the linear discriminant function in some feature space

$$f(\mathbf{x}) = (\mathbf{w} \cdot \Phi(\mathbf{x})) + b = \sum_{i=1}^{\ell} \alpha_i y_i (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x})) + b, \quad (1)$$

with $\mathbf{w}$ expressed as $\mathbf{w} = \sum_{i=1}^{\ell} \alpha_i y_i \Phi(\mathbf{x}_i)$. The quantity Φ denotes the mapping from input space $\mathcal{X}$ by explicitly transforming the data into a feature space $\mathcal{F}$ using $\Phi : \mathcal{X} \rightarrow \mathcal{F}$. (see Figure 1). SVM can do so implicitly. In order to train and classify, all that SVMs use are dot products of pairs of data points $\Phi(\mathbf{x}), \Phi(\mathbf{x}_i) \in \mathcal{F}$ in feature space (cf. Eq. (1)). Thus, we need only to supply a so-called kernel function that can compute these dot products. A kernel function k allows to implicitly define the feature space (Mercer's Theorem, e.g. [12]) via

$$k(\mathbf{x}, \mathbf{x}_i) = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}_i)). \quad (2)$$

By using different kernel functions, the SVM algorithm can construct a variety of learning machines, some of which coincide with classical architectures:

Polynomial classifiers of degree d: $k(\mathbf{x}, \mathbf{x}_i) = (\kappa \cdot (\mathbf{x} \cdot \mathbf{x}_i) + \Theta)^d$, where κ, Θ, and d are appropriate constants.

Neural networks(sigmoidal): $k(\mathbf{x}, \mathbf{x}_i) = \tanh(\kappa \cdot (\mathbf{x} \cdot \mathbf{x}_i) + \Theta)$, where κ and Θ are appropriate constants.

Radial basis function classifiers: $k(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{\sigma}\right)$, where σ is an appropriate constant.

Note that there is no need to use or know the form of Φ, because the mapping is never performed explicitly The introduction of Φ in the explanation above was for purely didactical and not algorithmical purposes. Therefore, we can computationally afford to work in implicitly very large (e.g. 10^{10}-dimensional) feature spaces. SVM can avoid overfitting by controlling the capacity and maximizing the margin. Simultaneously, SVMs learn which of the features implied by the kernel k are distinctive for the two classes, i.e. instead of finding well-suited features by ourselves (which can often be difficult), we can use the SVM to select them from an extremely rich feature space.

With respect to good generalization, it is often profitable to misclassify some outlying training data points in order to achieve a larger margin between the other training points (see Figure 1 for an example). This soft-margin strategy can also learn non-separable data. The trade-off between margin size and number of misclassified training points is then controlled by the regularization parameter C (softness of the margin). The following quadratic program (QP) (see e.g. [11], [13]):

$$\begin{array}{lll} \min & \|\mathbf{w}\|^2 + C \sum_{i=1}^{\ell} \xi_i & \\ \text{s.t.} & \rho(\mathbf{z}_i, f) \geqq 1 - \xi_i & \text{for all } 1 \leqq i \leqq \ell \\ & \xi_i \geqq 0 & \text{for all } 1 \leqq i \leqq \ell \end{array} \quad (3)$$

leads to the SV soft-margin solution allowing for some errors.

In this paper, we use VSMs, which are high dimensional

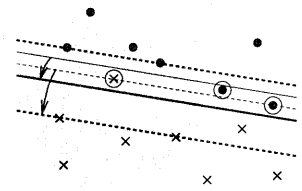

Fig. 1 A binary classification toy problem: This problem is to separate black circles from crosses. The shaded region consists of training examples, the other regions of test data. The training data can be separated with a margin indicated by the slim dashed line and the upper fat dashed line, implicating the slim solid line as discriminate function. Misclassifying one training example(a circled white circle) leads to a considerable extension(arrows) of the margin(fat dashed and solid lines) and this fat solid line can classify two test examples(circled black circles) correctlv.

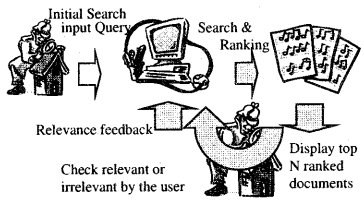

Fig. 2 Image of the relevance feedback documents retrieval: The gray arrow parts are made iteratively to retrieve useful documents for the user. This iteration is called feedback iteration in the information retrieval research area.

models, for the document retrieval. In this high dimension, it is easy to classify between relevant and irrelevant documents. Therefore, we generate the SV hard-margin solution by the following quadratic program.

$$\begin{array}{ll} \min & \|\mathbf{w}\|^2 \\ \text{s.t.} & \rho(\mathbf{z}_i, f) \geq 1 \quad \text{for all } 1 \leq i \leq \ell \end{array} \tag{4}$$

3. Active Learning with SVM in Document Retrieval

In this section, we describe the information retrieval system using relevance feedback with SVM from an active learning point of view, and several VSM representation of documents and several selecting rules, which determine displayed documents to a user for the relevance feedback.

3.1 Relevance Feedback Based on SVM

Fig. 2 shows the concept of the relevance feedback document retrieval. In Fig. 2, the iterative procedure is the gray arrows parts. The SVMs have a great ability to discriminate even if the training data is small. Consequently, we have proposed to apply SVMs as the classifier in the relevance feedback method. The retrieval steps of proposed method perform as follows:

Step 1: **Preparation of documents for the first feedback:** The conventional information retrieval system based on vector space model displays the top N ranked documents along with a request query to the user. In our method, the top N ranked documents are selected by using cosine distance between the request query vector and each document vector for the first feedback iteration.

Step 2: **Judgment of documents:** The user then classifiers these N documents into relevant or irrelevant. The relevant documents and the irrelevant documents are labeled. For instance, the relevant documents have "+1" label and the irrelevant documents have "-1" label after the user's judgment.

Step 3: **Determination of the optimal hyperplane:** The optimal hyperplane for classifying relevant and irrelevant documents is determined by using a SVM which is learned by labeled documents(see Figure 3).

Step 4: **Discrimination documents and information retrieval:** The documents, which are retrieved in the Step1, are mapped into the feature space. The SVM learned by the previous step classifies the documents as relevant or irrelevant. Then the system selects the documents based on the distance from the optimal hyper plane and the feature of the margin area. The detail of the selection rules are described in the next section. From the selected documents, the top N ranked documents, which are ranked using the distance from the optimal hyperplane, are shown to user as the information retrieval results of the system. If the number of feedback iterations is more than m, then go to next step. Otherwise, return to Step 2. The m is a maximal number of feedback iterations and is given by the user or the system.

Step 5: **Display of the final retrieved documents:** The retrieved documents are ranked by the distance between the documents and the hyper-plane which is the discriminant function determined by SVM. The retrieved documents are displayed based on this ranking(see Figure 4).

3.2 VSM Representations and Selection Rules of Displayed Documents

We discuss the issue of the term t_i in the document vector d_j. In the Information Retrieval research field, this term is called the term weighting, while in the machine learning research filed, this term is called the feature. t_i states something about word i in the document d_j. If this word is absent in the document d_j, t_i is zero. If the word is present in the document d_j, then there are several options. The first option is that this term just indicates whether this word i is present or not. This presentation is called boolean term weighting. The next option is that the term weight is a count of the number of times this word i occurs in this document d_j.

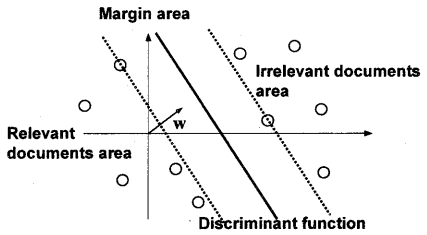

Fig. 3 Discriminant function for classifying relevant or irrelevant documents: Circles denote documents which are checked relevant or irrelevant by a user. The solid line denotes a discriminant function. The margin area is between dotted lines.

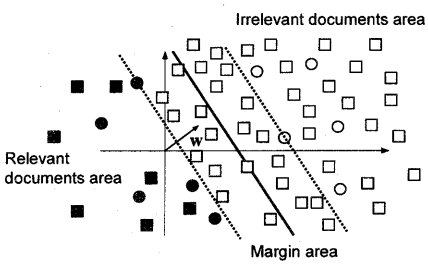

Fig. 4 Displayed documents as the result of document retrieval: Boxes denote non-checked documents which are mapped into the feature space. Circles denotes checked documents which are mapped into the feature space. The system displays the documents which are represented by black circles and boxes as the result of document retrieval to a user.

This presentation is called the term frequency(TF). In the original Rocchio algorithm [6], each term TF is multiplied by a term $\log\left(\frac{N}{n_i}\right)$ where N is the total number of documents in the collection and n_i is the number of documents in which this word i occurs. This last term is called the inverse document frequency(IDF). This representation is called the term frequency-the inverse document frequency(TFIDF) [1]. The Rocchio algorithm is the original relevance feedback method. In this paper, we compare the effectiveness of the document retrieval and the learning performance among boolean term weighting, term frequency(TF) and term frequency inverse document frequency(TFIDF) representations for our relevance feedback based on SVM.

Next, we discuss two selection rules for displayed documents, which are used for the judgment by the user. In this paper, we compare the effectiveness of the document retrieval and the learning performance among the following three selection rules for displayed documents.

Rule 1: The retrieved documents are mapped into the feature space. The learned SVM classifies the documents as relevant or irrelevant. The documents, which are discriminated relevant and in the margin area of SVM are selected.

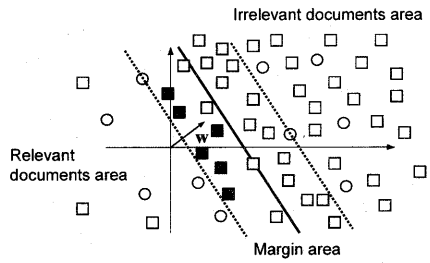

Fig. 5 Mapped non-checked documents into the feature space: Boxes denote non-checked documents which are mapped into the feature space. Circles denotes checked documents which are mapped into the feature space. Black and gray boxes are documents in the margin area. We show the documents which are represented by black boxes to a user for next iteration. These documents are in the margin area and near the relevant documents area.

From the selected documents, the top N ranked documents, which are ranked using the distance from the optimal hyperplane, are displayed to the user as the information retrieval results of the system(see Figure 5). This rule should make the best learning performance from an active learning point of view.

Rule 2: The retrieved documents are mapped into the feature space. The learned SVM classifies the documents as relevant or irrelevant. The documents, which are on the optimal hyperplane or near the optimal hyperplane of SVM, are selected. The system chooses the N documents in these selected documents and displays to the user as the information retrieval results of the system(see Figure 6). This rule is expected to achieve the most effective learning performance. This rule is our proposed one for the relevance feedback document retrieval.

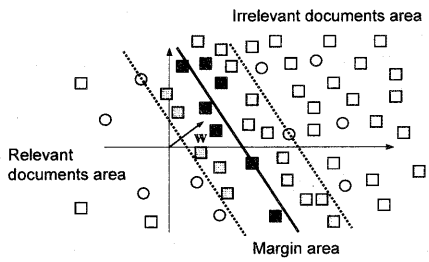

Fig. 6 Mapped non-checked documents into the feature space: Boxes denote non-checked documents which are mapped into the feature space. Circles denotes checked documents which are mapped into the feature space. Black and gray boxes are documents in the margin area. We show the documents which are represented by black boxes to a user for next iteration. These documents are near the optimal hyperplane.

4. Experiments

4.1 Experimental setting

In the reference [10], we already have shown that the utility of our interactive document retrieval with active learning of SVM is better than the Rocchio-based interactive document retrieval [6], which is conventional one. This paper presents the experiments for comparing the utility for the document retrieval among several VSM representations, and the effectiveness for the learning performance among the several selection rules, which choose the displayed documents to judge whether a document is relevant or irrelevant by the user. The document data set we used is a set of articles in the Los Angeles Times which is widely used in the document retrieval conference TREC [2]. The data set has about 130 thousands articles. The average number of words in a article is 526. This data set includes not only queries but also the relevant documents to each query. Thus we used the queries for experiments.

We adopted the boolean weighting, TF, and TFIDF as VSM representations. The detail of the boolean and TF weighting can be seen in the section 3.. And the detail of the adopted TFIDF can be seen in the reference [10]. In our experiments, we used two selection rules to estimate the effectiveness for the learning performance. The detail of these selection rules can be seen in the section 3..

The size N of retrieved and displayed documents at each iteration in the section 3. was set as twenty. The feedback iterations m were 1, 2, and 3. In order to investigate the influence of feedback iterations on accuracy of retrieval, we used plural feedback iterations. In our experiments, we used the linear kernel for SVM learning, and found a discriminant function for the SVM classifier in this feature space. The VSM of documents is high dimensional space. Therefore, in order to classify the labeled documents into relevant or irrelevant, we do not need to use the kernel trick and the regularization parameter C(see section 2.). We used LibSVM [14] as SVM software in our experiment.

In general, retrieval accuracy significantly depends on the number of the feedback iterations. Thus we changed feedback iterations for 1, 2, 3 and investigated the accuracy for each iteration. We utilized *precision* and *recall* for evaluating the two information retrieval methods [15] [16] and our approach.

4.2 Comparison of recall-precision performance curves among the boolean, TF and TFIDF weightings

In this section, we investigate the effectiveness for the document retrieval among the boolean, TF and TFIDF weightings, when the user judges the twenty higher ranked documents at each feedback iteration. In the first iteration, twenty higher ranked documents are retrieved using cosine distance between document vectors and a query vector in VSMs, which are represented by the boolean, TF and TFIDF weightings. The query vector is generated by a user's input of keywords. In the other iterations, the user does not need to input keywords for the information retrieval, and the user labels "+1" and "-1" as relevant and irrelevant documents respectively.

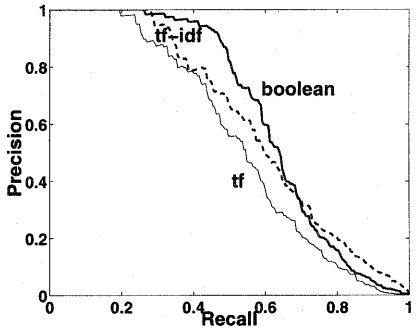

Fig. 7 The retrieval effectiveness of SVM based feedback(using the selection rule 2) for the boolean, TF, and TFIDF representations: The lines show recall-precision performance curve by using twenty feedback documents on the set of articles in the Los Angeles Times after 3 feedback iterations. The wide solid line is the boolean representation, the broken line is TFIDF representation, and the solid line is TF representation.

Figure 7 show a recall-precision performance curve of our SVM based method for the boolean, TF and TFIDF weightings, after four feedback iterations. Our SVM based method adopts the selection rule 2. The thick solid line is the boolean weighting, the broken line is the TFIDF weighting, and the thin solid line is the TF weighting.

This figure shows that the retrieval effectiveness of the boolean representation is higher than that of the other two representations, i.e., TF and TFIDF representations. Consequently, in this experiment, we conclude that the boolean weighting is a useful VSM representation for our proposed relevant feedback technique to improve the performance of the document retrieval.

4.3 Comparison of recall-precision performance curves between the selection rule 1 and 2

Here, we investigate the effectiveness for the document retrieval between the selection rule 1 and 2, which are described in the section 3..

Figure 8 show a recall-precision performance curves of the selection rule 1 and 2 for the boolean weightings, after four feedback iterations. The thin solid line is the selection rule 1, and the thick solid line is the selection rule 2. Table 1 gives

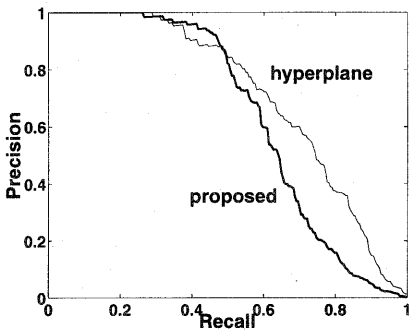

Fig. 8 The retrieval effectiveness of SVM based feedback for the selection rule 1 and 2: The lines show recall-precision performance curves by using twenty feedback documents on the set of articles in the Los Angeles Times after 3 feedback iterations. The thin solid line is the selection rule 1, and the thick solid line is the selection rule 2.

the average number of relevant documents in the twenty displayed documents for the selection rule 1 and 2 as a function of the number of iterations.

This figure shows that the precision-recall curve of the selection rule 1 is better than that of the selection rule 2. However, we can see from the table 1 that the average number of relevant documents in the twenty displayed documents for the selection rule 2 is higher than that of the selection rule 1 at each iteration. After all, the selection rule 1 is useful to totally put on the upper rank the documents, which relate to the user's interesting. When the selection rule 1 is adopted, the user have to see a lot of irrelevant documents at each iteration. The selection rule 2 is effective to immediately put on the upper rank the special documents, which relate to the user's interesting. When the selection rule 2 is adopted, the user do not need to see a lot of irrelevant documents at each iteration. However, it is hard for the rule 2 to immediately put on the upper rank all documents, which relate to the user's interesting. In the document retrieval, a user do not want to get all documents, which relate to the user's interest. The user wants to get some documents, which relate to the user's interest as soon as possible. Therefore, we conclude that the feature of the selection rule 2 is better than that of the selection rule 1 for the relevance feedback document retrieval.

5. Conclusion

In this paper, we adopt several representations of the Vector Space Model and several selecting rules of displayed documents at each iteration, and then show the comparison results of the effectiveness for the document retrieval in these several situations.

In our experiments, when we adopt our proposed SVM based relevance feedback document retrieval, the binary representation and the selection rule 2, where the documents that are discriminated relevant and in the margin area of SVM, are displayed to a user, show better performance of document retrieval. In future work, we will plan to analyze our experimental results theoretically.

Table 1 Average number of relevant documents in the twenty displayed documents for the selection rule 1 and 2 using the boolean representation

No. of feedback iterations	Ave. No. of relevant documents	
	selection rule 1	selection rule 2
1	7.125	11.750
2	7.750	9.125
3	7.375	8.875
4	5.375	8.875

References

[1] R. B. Yates and B. R. Neto. *Modern Information Retrieval.* Addison Wesley, 1999.

[2] TREC Web page. http://trec.nist.gov/.

[3] IREX. http://cs.nyu.edu/cs/projects/proteus/irex/.

[4] NTCIR. http://www.rd.nacsis.ac.jp/~ntcadm/.

[5] G. Salton and J. McGill. *Introduction to modern information retrieval.* McGraw-Hill, 1983.

[6] G. Salton, editor. *Relevance feedback in information retrieval*, pp. 313–323. Englewood Cliffs, N.J.: Prentice Hall, 1971.

[7] M. Okabe and S. Yamada. Interactive document retrieval with relational learning. In *Proceedings of the 16th ACM Symposium on Applied Computing*, pp. 27–31, 2001.

[8] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. In *Journal of Machine Learning Research*, Vol. 2, pp. 45–66, 2001.

[9] Harris Drucker, Behzad Shahrary, and David C. Gibbon. Relevance feedback using support vector machines. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 122–129, 2001.

[10] Takashi Onoda, Hiroshi Murata, and Seiji Yamada. Relevance feedback with active learning for document retrieval. In *Procedings of International Conference on Neural Networks 2003*, 2003.

[11] V.N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, 1995.

[12] B.E. Boser, I.M. Guyon, and V.N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *5th Annual ACM Workshop on COLT*, pp. 144–152, Pittsburgh, PA, 1992. ACM Press.

[13] B. Schölkopf, A. Smola, R. Williamson, and P.L. Bartlett. New support vector algorithms. Technical Report NC-TR-1998-031, Department of Computer Science, Royal Holloway, University of London, Egham, UK, 1998. *Neural Computation 2000.*

[14] Kernel-Machines. http://www.kernel-machines.org/.

[15] D. Lewis. Evaluating text categorization. In *Proceedings of Speech and Natural Language Workshop*, pp. 312–318, 1991.

[16] I. Witten, A. Moffat, and T. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images.* Van Nostrand Reinhold, New York, 1994.