Computational Linguistics and Chinese Language Processing
vol. 3, no. 2, August 1998, pp. 61-78

An Assessment of Character-based Chinese News Filtering Using Latent Semantic Indexing

Shih-Hung Wu*, Pey-Ching Yang*, Von-Wun Soo*

Abstract

We assess the Latent Semantic Indexing (LSI) approach to Chinese information filtering. In particular, the approach is for Chinese news filtering agents that use a character-based and hierarchical filtering scheme. The traditional vector space model is employed as an information filtering model, and each document is converted into a vector of weights of terms. Instead of using words as terms in the IR nominating tradition, terms refer to Chinese characters. LSI captures the semantic relationship between documents and Chinese characters. We use the Sin-gular-value Decomposition (SVD) technique to compress the term space into a lower dimension which achieves latent association between documents and terms. The results of experiments show that the recall and precision rates of Chinese news filtering using the character-based ap-proach incorporating the LSI technique are satisfactory.

1. Introduction

The rapid growth of the Internet has created the need of the Network Information Retrieval Systems. Most of the famous systems that assist people in locating information on the Internet, such as Lycos, Infoseek, Alta Vista, and WebWatcher [Armstrong 95], are designed for English information retrieval. To our knowledge, only the Csmart [Chien96] and GAIS [http://gais.cs.ccu.edu.tw/] systems are designed for Chinese information retrieval. However, information filtering is conceptually different from information retrieval, so we have to modify the techniques of information retrieval for

*Department of Computer Science, National Tsing Hua University, Hsin-Chu 30043 Taiwan R.O.C.
e-mail: shwu@cs.nthu.edu.tw, pcyang@cs.nthu.edu.tw, soo@cs.nthu.edu.tw

information filtering. In this paper, we assess the LSI technique for a hierarchical Chinese information filtering scheme. In particular, we assess the SVD approach to Chinese news filtering; to our knowledge, the SVD approach has never been investigated for the Chinese language.

Usenet news is one of the richest information resources on the Internet. Finding useful news among thousands of available news items is a crucial problem [Lang95]. Imagine a client user who needs a software agent to automatically recommend interesting news in Chinese from the Internet. Since news is updated every day, the traditional technique for information retrieval of using a fixed database will not work. Also, the task that a news filtering agent faces is to select relevant news, according to the user's interest or preference from a huge amount of dynamically growing news. Belkin and Croft [Belkin 92] pointed out that one major difference between information retrieval and filtering is that the queries in information retrieval typically represent the user's short-term interests while the user profiles in information filtering tend to represent the user's long-term interests. To model the user's long term interest, a user profile plays an important role in information retrieval [Mayeng 90] and filtering. Profiles can be represented in many ways and at different psychological and abstract levels. A collection of documents in a user's personal digital library may approximate the user profile. Information filtering is a document-find-document style of information retrieval. A document that is similar to the documents in the user's personal digital library is regarded as being relevant.

We adopt the vector space model [Yan 94] in our design of Chinese news filtering agents. In this model, each document is represented as a vector of weights of terms. We form each user profile by merging document vectors of the same interest category. The similarity of the incoming document vectors with the profile vectors can be computed by the means of cosine angles between the two vectors to determine if a document is to be filtered out.

In Chinese, there are no word delimiters to indicate the word boundaries; therefore, word segmentation is a difficult task. Many proper nouns or unknown words can not be found even in a word dictionary with a large vocabulary [Chien 95]. The number of different Chine-se characters is about 13,000, among which about 5,000 characters are the most commonly used characters. However, the number of Chinese words in a document collection set can easily be up to 1,000,000. To represent a personal profile in terms of words, the difficulty of word segmentation in Chinese must be dealt with [Chien 96]. We will show through experi-ments that without word segmentation, characterbased filtering incorporating LSI can be a satisfactory information filtering method.

The filtering method incorporating with LSI selects relevant documents whose contents have no exactly matched keywords. This is quite different from traditional techniques, such as the Boolean models. The probabilistic model, the Bayesian Belief Network Model [Turtle 91] [Ribeiro 96] shares similar feature. The Boolean models exactly match the document's terms with the combination of the search terms specified in the query. The probabilistic models estimate the degree of relevance between documents and a user query by considering the appearance frequency of certain terms in the documents and the user query, together with information about term distribution in the document collection.

Since individual terms and keywords are not adequate discriminators of the semantic content of documents and queries, the performance of the conventional retrieval models often suffers due either to missing relevant documents which are not indexed by the keywords specified in the query, or to retrieval of irrelevant documents which are indexed based on an unintended sense of the keywords in the query. Therefore, there has been great interest in text retrieval research that is based on semantics matching instead of strictly keyword matching.

Latent Semantic Indexing (LSI) using Singular-value Decomposition (SVD) is one approach to overcoming this deficiency of exact keyword matching techniques. We use truncated SVD to capture the semantic structure of the word usage in certain documents and hope that this usage can be applied to other documents. Using the singular value matrix from the truncated SVD, a high-dimensional vector space representing a term-document matrix is mapped to a lower dimension matrix that reflects the major concept factors in the specified documents while ignoring the less important ones. Terms occurring in similar documents will be nearer in the reduced vector space. With LSI, documents may satisfy a user's query when they share terms that are closer to each other in the reduced space. Since the reduced vector spaces are more robust indicators of the semantic meaning than are individual words, the performance may be better than that of the original space.

Several papers have reported the use of the LSI method. An example was assigning submitted manuscripts to the reviewers of the Hypertex'91 conference based on the interests of each reviewer; using the LSI method, a set of relevant manuscripts was sent to the reviewer [Dumais 92]. The automated assignment method achieved good matching between the reviewers and their interests just as did assignment by the human experts. Syu presented the technique of incorporating LSI into a neural network model for text retrieval [Syu96]. The performance, in terms of precision and recall, was comparable to that of the text retrieval model.

The remainder of this paper is organized as follows: Section 2 provides an overview of the Latent Semantic Indexing method applied to information retrieval, and how truncated SVD can be used as an LSI approach. Section 3 briefly reviews our information filtering scheme. Section 4 reports experimental results obtained using the LSI-based model, and Sec-tion 5 offers the discussion and a conclusion.

2. Latent Semantic Indexing method applied to information retrieval

Latent Semantic Indexing (LSI) is an extension of the vector space retrieval method. We assume that there is some unknown "Latent" association in the pattern of terms or keywords used among documents [Dumais 92], and try to estimate this latent association. Singular-Value Decomposition (SVD) is a technique for eigenvector decomposition and factor analysis used in statistics [Cullum 85], and Latent Semantic Indexing (LSI) using SVD is one approach used to model the latent semantic relationship between the documents and the index terms. This approach performs singular-value decomposition on a term-by-document matrix, thus generating a reduced space with lower dimension. The similarity between two documents is calculated according to the index terms used in each of the documents that occurr in other documents. Using the LSI representation, the documents satisfy a user query when they share terms of similar semantic meaning in the reduced vector space. The dimension of the resulting vector space is much smaller than the number of exact index terms used in a document collection (e.g., from several thousands to 100 or 300 [Dumais 94]), and a filtering model using LSI can save time and memory.

2.1 Singular-Value Decomposition (SVD) and truncated SVD

SVD is a reliable tool for matrix factorization. For any matrix A, A^TA has nonnegative eigen-values. The nonnegative square roots of the eigenvalues of A^TA are called the singular values of A, and the number of non-zero singular values is equal to the rank of A, *rank (A)*. Assume that A is an m *by* n matrix and that *rank (A)* = r; the singular -value decomposition of A is de-fined as

$$A = U W V^T,$$

where the size of U is m *by* m, the size of V is n *by* n, and the size of W is m *by* n. Both U and V^T are orthogonal matrices, i.e., $UU^T = I_m$, and $VV^T = I_n$; W is a diagonal matrix consisting of the singular values of A: $\sigma_1, \sigma_2, \cdots, \sigma_r$. And the σ_j's are the singular values of A, $\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_r \geqq 0$, and $\sigma_j = 0$ for $j \geqq r+1$.

To apply SVD as an LSI tool, a term-by-document matrix A must be constructed. Then we can use SVD to generate the optimal approximation of the document representation specified by the matrix A. Since the singular values in the matrix W are ordered from largest to smallest, the first largest k singular values may be kept, and the remaining smaller ones are set to zero. As a result, the representations of the matrices U, V, and W can be reduced to obtain a new diagonal matrix W_k by removing columns and rows which are zeros from W; a matrix U_k can be obtained by removing the $(k+1)$st to the mth columns from U; and a matrix V_k can be obtained removing the $(k+1)$st to the nth rows from V. The product of the resulting matrices is a matrix A_k which is an approximation of the matrix A, and rank $(A_k)=k$:

$$A_k = U_k W_k V_k^T,$$

The relation between matrices A and A_k is shown in **Figure 1**. The LSI method using SVD can be viewed as a technique for deriving a set of non-correlated indexing of factors (i.e., the singular vectors) which represent different concepts in the usage of words in the document collection. The documents and queries are then represented by the vectors of factor values instead of the individual index terms. Using the k-largest factors, it may be possible to capture the most important latent semantic relation between documents and index terms, and to avoid unintended sense in word usage.

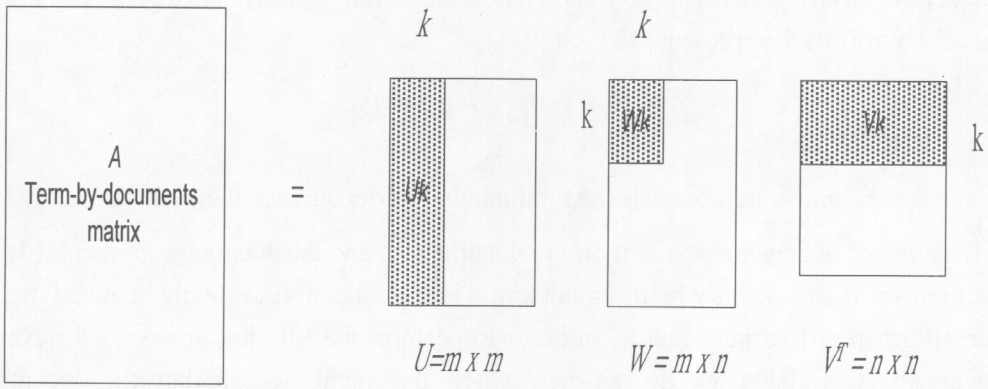

Figure 1 *An illustration of truncated SVD of a term-by-document matrix A.*

2.2 Document and query representations

Since the term-by-document matrix A has been reduced to a lower dimension matrix A_k, the vector which represents the query and all the incoming documents must be mapped to the same dimension of the matrix A_k. Using singular-value decomposition, a term-by-document matrix A is mapped into a reduced k by n matrix represented by $W_k V_k^T$, which relates k factors to n documents. A query q, originally of dimension size m, can be mapped into a size k vector q' :

$$q' = (q^T U_k W_k^{-1})^T.$$

The similarity between two documents is then computed using this shorter vector representation.

3. The character-based Chinese news filtering Scheme

3.1 The character-based vector representation of documents and personal profiles

A Chinese character is the basic processing unit and is used much like as the concept of a "term" in the IR denominating tradition; we use terms to refer to Chinese characters in this paper. In our approach, we need no stemming, no stop word list and no thesaurus. We represent the weight of a term in a given document by adopting Salton's well-tested *TFIDF* for-mula in IR, the term frequency (tf) multiplied by the inverse document frequency (idf) [Salton 89] [Salton 91]. That is, the weight of a term t in a given document d, namely w (t,d), is represented as

$$w (t, d) = tf_{t,d} * log (N/df_t),$$

where the document number N is the total number of documents, the term frequency $tf_{t,d}$ is the number of appearances of term t in document d, and the document frequency dft is the number of documents which content term t in the collection. It should be noted that in our information filtering scheme, in order to perform the filtering process whenever a document is available, we do not collect new documents to calculate the document frequency. Instead, we use a fixed set of document to represent the documents frequency for all the incoming documents.

A document *D* can be represented as a vector *V* with elements *v1, v2, ⋯, vn*, where n is equal to the size of the character vocabulary, and *vi* is the weight of term i in the document. All vectors are normalized for convenience and by convention. We can calculate the similarity between two documents *Di* and *Dj* by means of the cosine of the angle between their vector representations:

$$\text{Similarity } (Di, Dj) = \frac{Vi * Vj}{\|Vi\| \|Vj\|} ,$$

where V_i is the vector representation of *Di*; * represents the inner product between two vectors; $\|Vi\|$ represents the norm of a vector *Vi*. Based on the formula, two documents with the same character set will have the highest value of similarity between them because the inner product of the two document vectors will be one while two documents without any character in common will have the lowest similarity: zero.

We merge the document vectors in the same interest group (grouped either by the user or by a classification/clustering agent) into a higher level profile vector by summing up their vector. The profile vector is also normalized.

3.2 The tasks of a news filtering agent

A Chinese character-based news filtering agent will carry with it a set of weights in terms of inverse document frequencies as discussed above for a vocabulary of terms (characters), and a profile vector that represents a certain interest category and a similarity threshold associated with the profile vector. For each news document in the news server, the filtering agent will convert it first into a document vector; then, the similarity between the document vectors and the profile vectors will be computed according to the similarity measurement method discussed in section 3.1. If the similarity of the document with the profile is lower than the threshold, it is filtered out. In order to calculate every different recall rate, we treat the thresh-old as a variable in the experiment.

3.3 The hierarchical information filtering scheme

The hierarchical information filtering scheme [Soo 97] reduces the agent's total task. By com-posing profile vectors, we reduce the number of vectors that each agent must compare with document vectors on the Web. A user picks some interesting documents, and then the system converts the documents into vectors and merges the vectors into one profile vector. As the number of profile vectors increases (each representing a group of interest to the user), the lower-level profile vectors are combined to form higher-level

profile vectors. We may assume that the final highest level profile vector can represent an overall interest of the user. The intelligent news filtering agent can then use this profile vector to search for relevant documents on the Web. All the documents that pass the higher-level filtering process are sent to the lower-level filter. Previous work showed that the hierarchical information filtering scheme can save computation time while maintaining the same recall-precision rate [Soo 97].

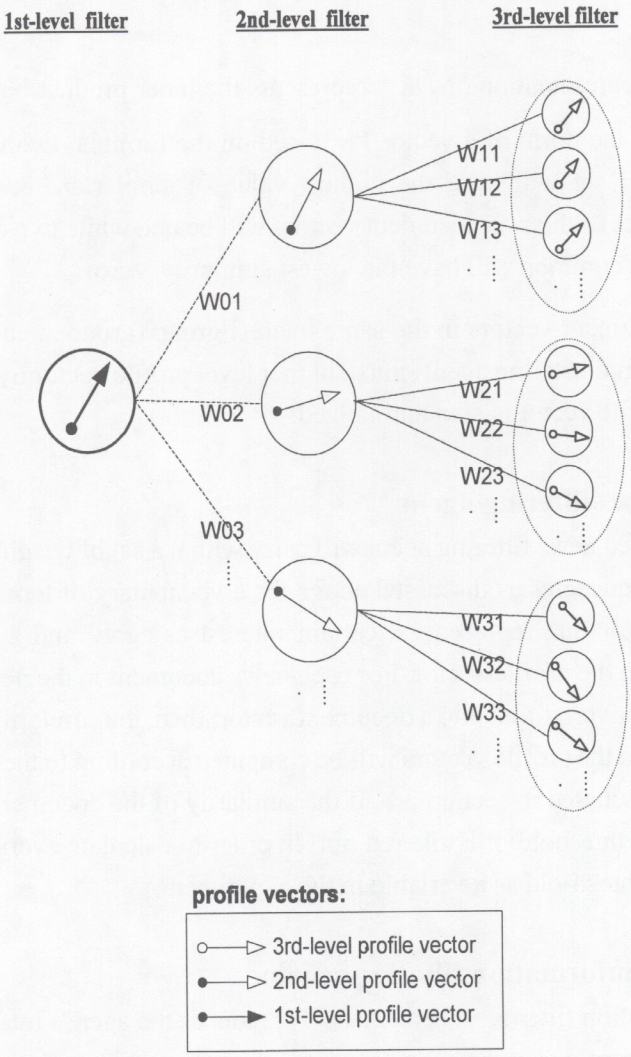

Figure 2 *The hierarchical information filtering scheme.*

4. Experimentation

Since the objective answer to the question of whether a document is relevant to the profile is not available, we tried to use the categories given by the news press to objectively form the groups of different interests to the user. Then, we assessed our method. The software package used to solve the SVD was MATLAB.

4.1 Data collection and document vectorization

We gathered three sets of articles from the on-line China Times [http://www.chinatimes.com.tw/] during 2 consecutive weeks from Mar.2nd 1997 to Mar.15th 1997. There were 671 articles in the first week and 669 in the second. These articles were written by professional reporters, and we collected articles from all the categories that the China Times provided. The categories were: *Entertainment, Sports, Economy, Focus, International, Mainland, Social, Taiwan and Editorial.*

Table 1. *The number of documents in the document collection sets.*

Category \ Set	1st week	2nd week
Economy	86	95
Editorial	14	14
Entertainment	80	82
Focus	80	79
International	70	63
Mainland	63	58
Social	67	73
Sports	90	87
Taiwan	114	111
Total	671	669

Table 1 shows the number of documents in each of the categories. The length of each article was about 500-2000 Chinese characters. In order to test wheather the frequency of words in the document collection sets was stable or not, we used the 671 articles in the first week as the training set to compute the document frequency *dft* for each term *t* (see **Figure 3**)and the 669 articles in the second week as the testing set for the filtering experiment.

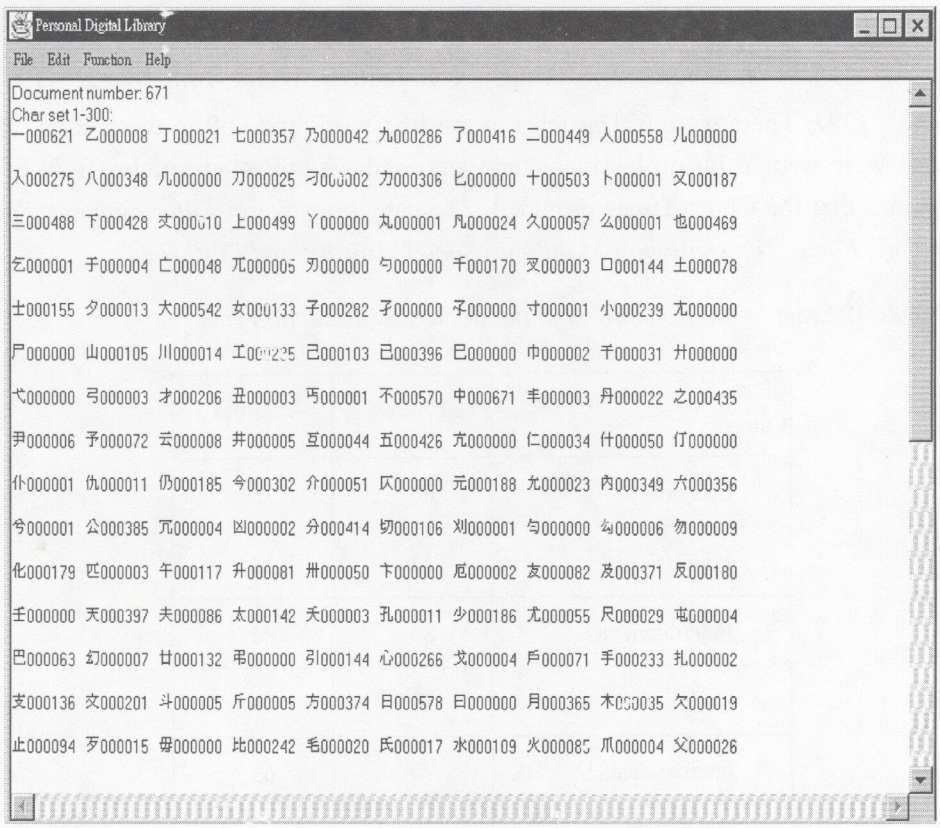

Figure 3 *A partial view of the document frequency dft for each term t.*

The articles were first transformed into normalized document vectors as discussed in section 2.1; all English characters and Arabic numerals were ignored. The similarity between two documents was then equal to the inner product of two vectors. **Figure 4** shows the similarity graph of the testing data; each document compared to all six hundred documents . The six hundred documents were selected from the document collection and ordered from 1 to 600 according to the categories: Economy, Editorial, Entertainment, Focus, International, Mainland, Social, Sports, and Taiwan, respectively; as in **Table 1**.

The gray levels represent the similarity values between two documents; the darkest point is of the highest similarity value. We can see that documents in the same category tended to have higher similarity values. This can be inferred from the fact that in the similarity graph, the darker points seem to form a symmetrical region along the diagonal line.

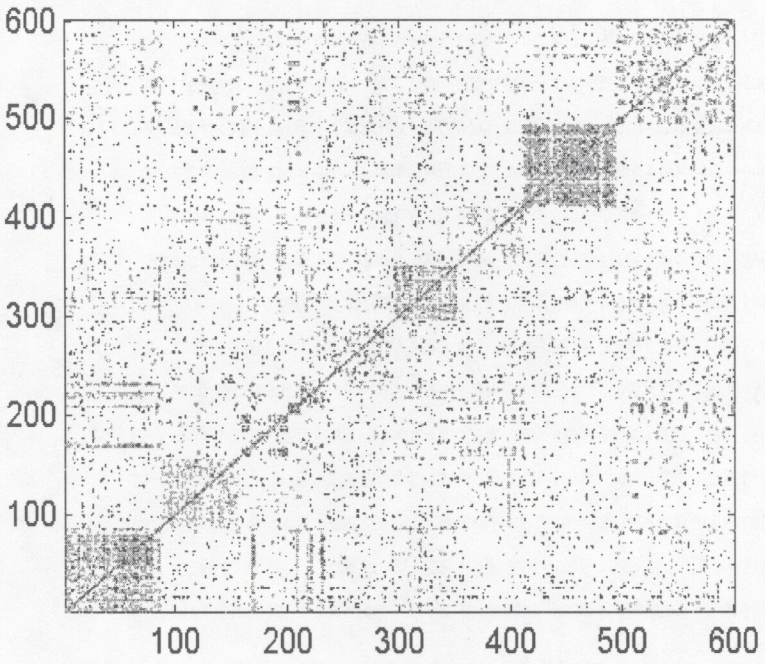

Figure 4 *The similarity graph of the testing data. Each document compared to all six hundred documents.*

4.2 Experiments on information filtering with SVD

To mimic a user's interests, we randomly choose news articles from three categories (Entertainment, Sports, and Economy) on the same day (Mar. 2nd) to form the initial user's profiles. A user profile can be treated as a set of documents which are transformed into normalized document vectors as shown in **Figure 5**.

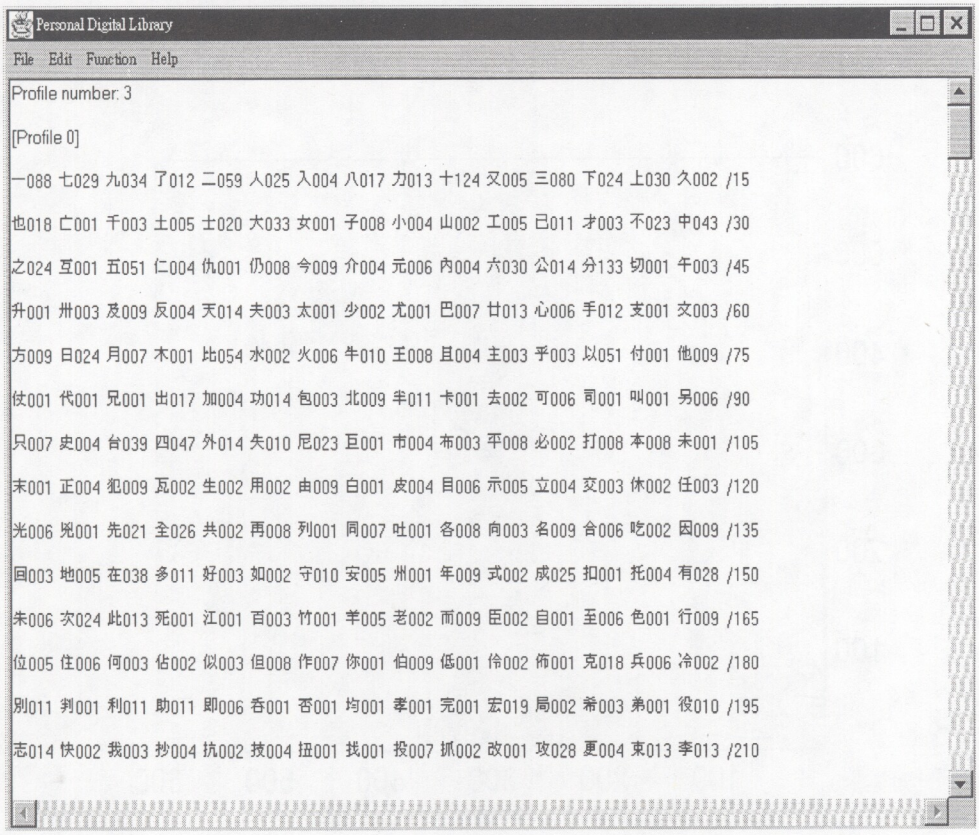

Figure 5 *A partial view of the profile vectors in the experiment.*

To evaluate the effectiveness of news filtering based on the character-based method for Chinese news document, the tf-idf weighting and vector space model were adopted in our experiment on the nine news categories, and we tried different k values when applying truncated SVD. As discussed above, in our hierarchical information filtering scheme, several arbitrary articles from each category in the training set were selected and merged into one profile document. The profile document was then transformed into a normalized vector, called a profile vector. By comparing the profile and document vectors in the test set, we could retrieve the most similar documents in the test set and measure the precision against different recall values as plotted in **Figures 6-8**.

From **Figures 6-8**, we observe that different values of *k* in SVD had different performance. The performance was worse either when the *k* value was small, e.g., 2, or when the *k* value was large, e.g., 100. The experiments showed taht the suitable value for *k* is 10 for our document collection sets. Dumais suggested that the probable *k* value is from 100 to 300 for an English document [Dumais 94]. The difference may come form the different language stuctures of Chinese and English. However, the result is what we want. The great reduction of the vector dimension saves a lot of memory space and time needed to further utilize the document vectors. We performed more experiments to justify our observation.

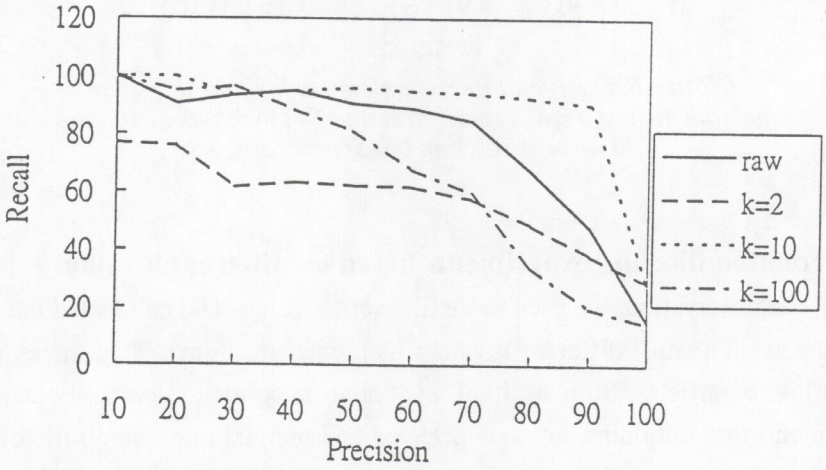

Figure 6 *The Recall-precision curves of four different methods (using a raw vector form and SVD with k=2, 10, and 100, respectively) on the Economy category.*

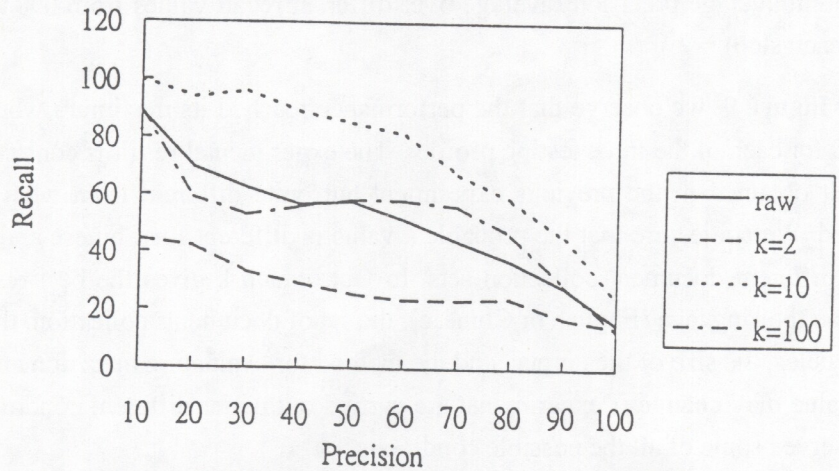

Figure 7 *The recall-precision curves for four different methods (using a raw vector form and SVD with k=2, 10, and 100, respectively) on the Entertainment category.*

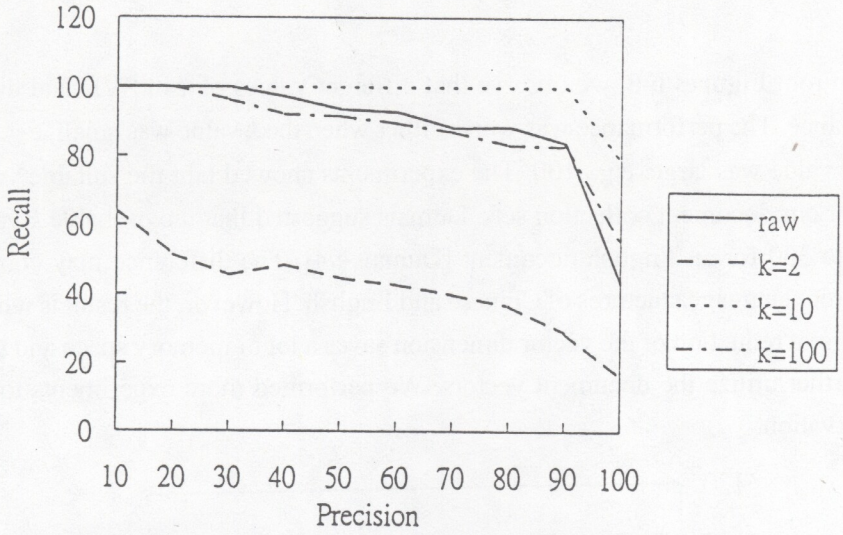

Figure 8 *The recall-precision curves for four different mehtods (using a raw vector form and SVD with k=2, 10, and 100 respectively) on the Sports category.*

4.3 Information filtering experiments based on different k values

To justify our observation, we tried more different k values and calculated the 11-point average precision against different k values as plotted in **Figure 9**. In the experiment, several arbitrary articles from each of the three categories (Sports, Economy, and Entertainment) in the training set were selected and merged into one profile document. The profile document was then transformed into a normalized vector named as a profile vector. By comparing the profile vector and document vectors in the test set, we could retrieve the most similar documents in the test set and measure the performance based on the 11-point average precision (average over different recall values from 0% to 100%, 10% in each step).

1. From **Figure 9**, we observe that the performance reached its maximum when k was about 10 for each of the three testing profiles. The experimental result is consistent with the result obtained in the previous experiment but quite different from what Dumais suggested. We conjecture that the probable *k* value is different for Chinese and English and for different document collection sets. In fact, which k gives the best result may depend on the language (English or Chinese), the set of documents collection, the length of the articles, the size of the corpus, and the method for evaluating precision and recall. The k value may change if experiments are carried out under different conditions. We only observed some of all the possible conditions.

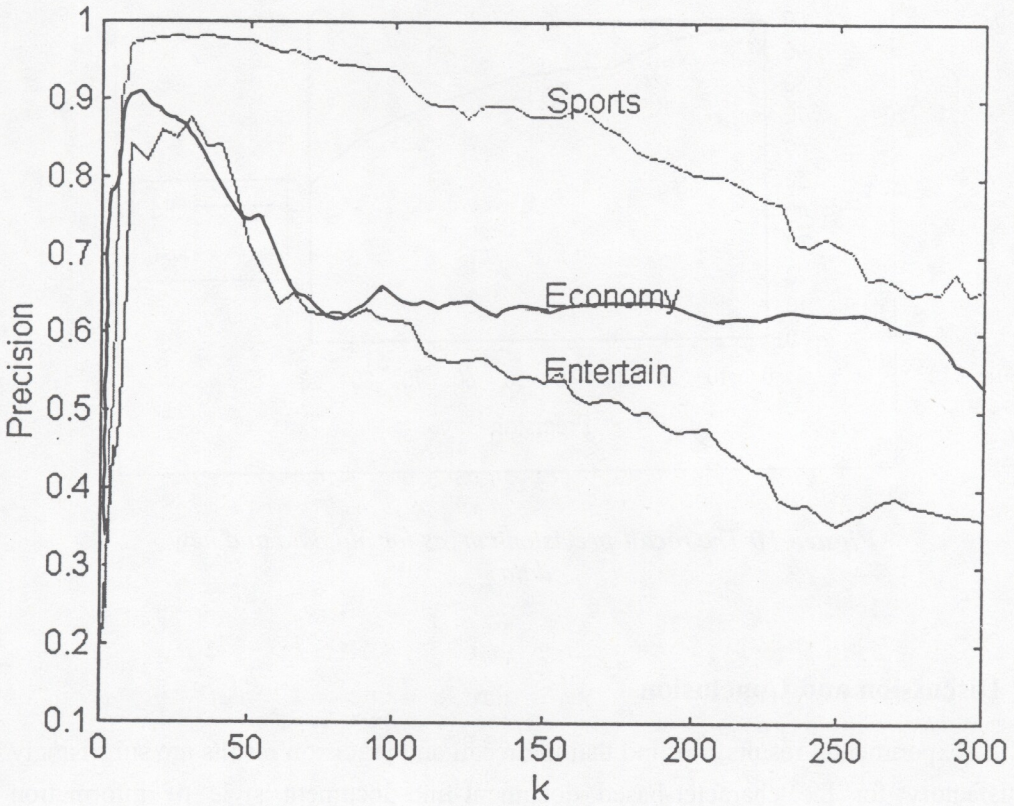

Figure 9 *Performance (11-points average precision) variation against different k*

4.4 Experiments on the testing data collected one year later

In order to find out if the filtering method is stable or not, we performed the same experiments one year later. We collected 1190 documents from the same WWW site. This time, we repeat the filtering process on the new testing data without changing our training data and profiles. The recall-precision performance decreased a little bit, namely, to 4.2% on average (see Figure 10).

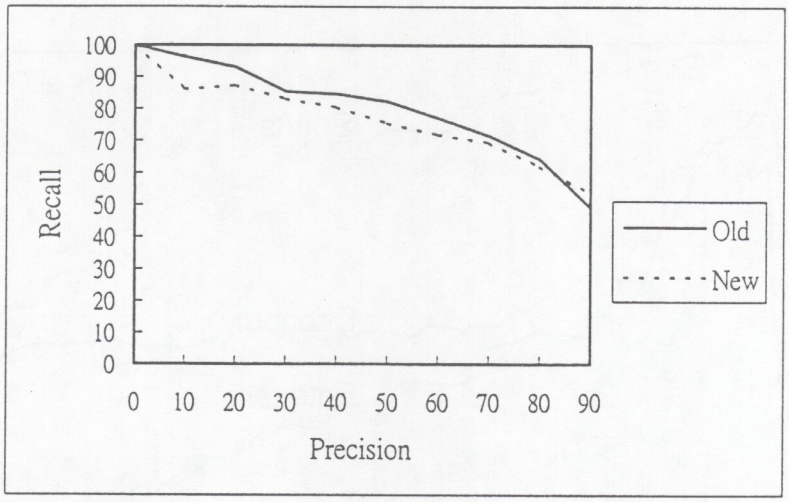

Figure 10 *The recall-precision curves for old data and new data*

5. Discussion and Conclusion

In the experimental results, we find that the recall and precision results are surprisingly satisfactory for the character-based document-find-document style of information filtering of Chinese news. The SVD technique can be used to reduce the storage space need of the term-by-document matrix and the processing time needed for further utilization. The difference in the performance for different choices of k with the truncated SVD value is quite interesting.

The results show that articles with similar character sets tend to have similar meaning, and that the semantic meaning of a Chinese news article can be, to some degree, implied by it's character set. Even though in Chinese, different orders of the same set of characters may have different meanings, and the same word may have ambiguities in parts of speech, character-based filtering can work well with the information provided by the context of an article.

The character-based information filtering scheme makes a lot of sense because no dictionary is rich enough that can contain all the possible words, and because the word segmentation task in Chinese is difficult. Only the weights and counts of the most commonly used characters in a documents collection set are needed to design an intelligent news filtering agent. A compressed matrix representation yields better performance and saves more computation time for the filtering agents. The SVD method can reduce the size of the term-by-document matrix and sort the significance of dimensions

for the matrix. This is why a suitable choice of k will give better performance. The first k dimensions are necessary and sufficient to discriminate the categories. If we view stop words as noise, the larger the k value, the more noise will be considered. On the other hand, a small k may be insufficient for discrimination among categories. Several experiments in Chinese IR have shown that single-character based indexing cannot provide effective results when dealing with large size texts [Kwok 97]. However, in our domain, the news articles tended to be short.

Representing the user profile and performing news filtering hierarchically not only has the merit of reducing the computation cost, but also has potential for performing the information filtering task in a distributed and parallel manner. The efficiency will be improved even more if each profile vector runs independently on a distributed system. This could be achieved because of the independence property of the profile and document vectors; i.e., they do not interfere with each other when similar calculations are performed.

In the future, relevance feedback from the user can be used to improve the performance by adjusting several system parameters. It can be used to adjust the thresholds in each stage or to adjust the weights to combine lower level profile vectors into higher level ones. In this direction, we now are studying into machine learning techniques as neural networks [Pannu 95] [Syu 96].

Acknowledgment

This work was financially supported by the Institute for Information Industry and the National Science Council of Taiwan, Republic of China, under grant No. NSC86-2213-E-007-53.

References

Armstrong, R. and D. Freitag, T. Joachims, and T. Mitchell, "WebWatcher: A learning apprentice for the world wide web", *1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, Stanford, March 1995.

Belkin, N.J. and Croft, W.B., "Information filtering and information retrieval: two sides of the same coin?", Comm. ACM 35, 12 (Dec.), pp. 29-38.

Chien, L.F., "Fast and quasi-natural language search for gigabytes of Chinese texts", In *Proceedings of 18th ACM SIGIR*, pp.112-120, 1995.

Chien, L.F., "An intelligent Chinese information retrieval system for the Internet", In *Proceedings of the ROCLING IX*, 1996.

Cullum, J.K. and R.A. Willoughby, "Lanczos Algorithms for Large Symmetric Eigen value

Computations - Vol. 1, Theory (Ch 5: Real Rectangular Matrices)", Brikhauser, Boston, 1985.

Dumais, S.T. and J. Nielsen, "Automating the Assignment of Submitted Manuscripts to Reviewers", In *Proceedings of the 15th ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 233-244, 1992.

Dumais, S.T., "Latent Semantic Indexing and TREC-2", *The Second Text Retrieval Conference (TREC-2)*, NIST Special Publication 500-215, pp. 105-115, 1994.

Kwok, K.L., "Comparing Representations in Chinese Information Retrieval", In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp. 34-41, 1997.

Lang, K., "Newsweeder: learning to filter Netnews", *Proceedings of the Twelfth International Conference on Machine Learning*, 1995.

Mayeng, S. H. and R. R. Korfhage, "Integration of user profiles: models and experiments in information retrieval. Information Processing and Management", Vol. 26, No. 6, 1990.

Ribeiro, B.A.N. and R. Muntz, "A Belief Network Model for IR", In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, pp.253-269, 1996.

Salton, G., "Automatic Text Processing", Addison Wesley, Reading, Massachusetts, 1989.

Salton, G., "Developments in automatic text retrieval", *Science 253*, 1991.

Pannu, A. S. and K. Sycara, "A learning personal agent for text filtering and notification", *Proceedings of the International Conference of Knowledge Based Systems (KBCS 96)*, Dec. 1996.

Soo, V.W., P.C. Yang, S. H. Wu and S.Y. Yang, "A Character-based Hierarchical Information Filtering Scheme for Chinese News Filtering Agents", *Proceedings of the Second International Workshop on Information Retrieval with Asian Languages (IRAL-97)*, Oct. 1997.

Syu, I., S. D. Lang, and N. Deo, "Incorporating latent semantic indexing into a neural network model for information retrieval", *Proceedings of the Fifth International Conference on Infor-mation and Knowledge Management*, Nov. 1996.

Turtle, H. and W. B. Croft, "Evaluation of an Inference Network based Retrieval Model", *ACM Transactions on Information Systems*, Vol. 9, No. 3, July 1991.

Yan, T. W. and H. Garcia-Molina, "Index structures for information filtering under the vector space model", *Technical Report STAN CS-TR-93-1494*, Nov. 1993.