A Text Mining Model by Using Weighting Technology

Chenggen Shi
Faculty of Information Technology
University of Technology, Sydney
Po Box 123, Broadway, NSW 2007, Australia
cshi@it.uts.edu.au

Jie Lu
Faculty of Information Technology
University of Technology, Sydney
Po Box 123, Broadway, NSW 2007, Australia
jielu@it.uts.edu.au

ABSTRACT

In Latent Semantic Indexing (LSI) has been proven to be a valuable analysis tool with a wide range of applications. However choosing an appropriate number of dimensions for LSI is still a crucial challenge. This paper provides a document vector model, by using weighting technology, to deal with this problem. Our experimental results have demonstrated that this model can detect a dataset structure, help determine an appropriate number of dimensions for LSI.

Keywords

Latent semantic indexing, Multi-objective, Information retrieval, Text mining, Weighting technology.

INTRODUCTION

As digital libraries and the World–Wide–Web (WWW) continue to proliferate the enormous volume of online textual material, developing intelligent information retrieval technology becomes one of the great challenges. Latent Semantic Indexing (LSI) (Deerwester et al. 1990) is an approach to automatic indexing and information retrieval. The general claim is that similarities between documents or between documents and queries can be more reliably estimated in the reduced latent space representation than in the original representation. Although LSI has been applied with remarkable success in different domains (Foltz et al. 1992, Landauer et al. 1997, Bellegarda et al. 1998, Dumais et al. 1998, Cristianini et al. 2002, Lanauer et al. 1997, Rehder et al. 1998, Wiemer et al. 1999, Wiemer 2000 and Kawamae et al 2002), its theoretical foundation remains to a large extent unsatisfactory and incomplete. The principle challenge is how to choose an appropriate number of dimensions for the LSI (Deerwester et al. 1990). One can intuitively say that the appropriate number of dimensions is large enough to fit all the real structure in the data, but small enough so that we do not also fit the sampling error or unimportant details. If too many dimensions are used, the method begins to approximate standard vector methods and loses its power to represent the similarity between words. If too few dimensions are used, there is not enough discrimination among similar words and documents or it will create a serious distortion of word to word and document to document similarity.

The primary goal of this paper is to present a novel model to choose an appropriate number of dimensions for the LSI given a dataset. Our experimental results show that the proposed approach is effective and promising. The rest of this paper is organized as follows. Section 2 overviews LSI technologies and presents document vector linear association model. Section 3 describes experimental results and observation about document vector linear association and the number of dimensions. Section 4 addresses performance evaluations of the proposed model. The conclusions are summarized in the last section 5.

This paper provides a document vector model which can be used to choose the appropriate number of dimensions for LSI. Our experimental results demonstrate that the proposed model is effective and promising. The remainder of this paper is organized as follows. The background and a document vector model are addressed in Section 2. Section 3 describes experimental results. Conclusions are summarized in Section 4.

BACKGROUND AND A DOCUMENT VECTOR MODEL

Overview of Latent Semantic Indexing Technology

A key idea of LSI is to map word-document to a vector space of reduced dimensionality, the latent semantic space (Deerwester et al. 1990). The mapping is restricted to be linear and based on a Singular Value Decomposition (SVD) of a co–occurrence table. The singular value decomposition (Cullum et al. 1985 and Gene et al. 1981) is commonly used in the solution of unconstrained linear least squares problems, matrix rank estimations, and canonical correlation analysis. Given a $M \times N$ matrix $A = (a_{mn})_{M \times N}$, where $M \geq N$, the SVD of matrix A is given by (Deerwester et al. 1990):

$$A = U\Sigma V^T \qquad (1)$$

Where U is an $M \times N$ orthogonal matrix and $U^T U = I$; V is an $N \times N$ orthogonal matrix and $V^T V = I$; Σ is a $N \times N$ diagonal matrix with positive or zero elements (the singular values). For $h = 1, \ldots N$, let σ_h be a singular value of A. The singular values are ordered, so that

$$\sigma_1 \geq \sigma_2 \ldots \geq \sigma_N \qquad (2)$$

The LSA approximation of the matrix A is computed by setting all but the largest K singular values in Σ to zero. An approximation $\widetilde{A}$ for A is given by:

$$\widetilde{A} = U\widetilde{\Sigma}V^T \qquad (3)$$

where $\widetilde{\Sigma}$ is the approximations of Σ.

Notice that the document-to-document dot products based on this approximation are given by :

$$\widetilde{A}^T \widetilde{A} = (U\widetilde{\Sigma}V^T)^T U\widetilde{\Sigma}V^T = (V\widetilde{\Sigma}U^T)(U\widetilde{\Sigma}V^T) = V\widetilde{\Sigma}^2 V^T \qquad (4)$$

While the original high dimensional vectors are sparse, the corresponding low dimensional latent vectors will typically not be sparse. This implies that it is possible to compute meaningful association values between pairs of documents, even if the documents do not have any words in common.

A Document Vector Model

In general, for $A = U\Sigma V^T$, the matrices U, Σ and V must be of full rank [1]. The beauty of a SVD is to allow a simple strategy for optimal approximate fit using smaller matrices. According to (2) the singular values in Σ are ordered by size, so the first K largest may be kept and the remaining smaller ones set to zero (= $\widetilde{\Sigma}$). Since zeros were introduced into $\widetilde{\Sigma}$, the representation can be simplified by deleting the zero rows and columns of $\widetilde{\Sigma}$, to obtain a new diagonal matrix Σ_K, and then deleting the corresponding columns of U and V to obtain U_K and V_K respectively. Actually, we approximate the matrix A keeping only the first K largest singular values and the corresponding columns from U and V. The result is described into a reduced matrix: A_K, a $M \times N$ matrix which is constructed from the K-largest singular triplets of the matrix A,

$$A_K = U_K \Sigma_K V_K^T \qquad (7)$$

where U_K, a $M \times K$ matrix, is the first K columns of the U; V_K, a $N \times K$ matrix, is the first K columns of the V; Σ_K, a $K \times K$ matrix, keeps only the first K singular values from Σ.

A dot product $\Phi_{ij}(K)$ $i = 1, \ldots, M, j = i+1, \ldots, M, K = 1, \ldots, N$ between any two document vectors of the matrix A_K is defined as

$$\Phi_{ij}(K) = \sum_{m=1}^{M} a_{mi}^K a_{mj}^K \qquad (8)$$

where $a_{ij}^K \in A_K$.

By using (8), a document meaningful association value between any pair of document can be computed. When K varies over the set $\{1, \ldots, N\}$, $\Phi_{ij}(k)$ varies as well. The maximum of $\Phi_{ij}(K)$ is defined as

$$\max\{\Phi_{ij}(K), K \in 1, \ldots, N\} \qquad (9)$$

The main task of LSI is to deal with the synonymy problem. If a dot product between two document vectors has a larger value, these two documents have stronger similarity. The LSI will achieve the best performance, if we choose a particular value of K to satisfy (9) at the same time. Maximizing all the dot products among all document vectors is a multi-objective maximization problem. In general, simultaneous maximizing multiple objective functions is not always possible when the objective functions conflict with each other. By using a weighting method proposed by Sakawa (Sakawa 1993), we can obtain a compromise solution for (9). By taking the weighted sum of all the objective functions of (9), we have

$$\Phi(K_0) = \max\{\Phi(K), K = 1,\ldots,N\} = \max\{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} w_{ij}\Phi_{ij}(K), K \in 1,\ldots,N\} \quad (10)$$

where $w_{ij} \in (0,1)$ is a weighting coefficient of Φ_{ij}, $i = 1,\ldots,N-1, j = i+1,\ldots,N$. $\Phi(K)$ is called a document vector linear association for K, $K = 1,\ldots,N$. When all w_{ij} are fixed, a compromised solution K_0, that corresponds to $\Phi(K_0)$, is obtained. That means that an appropriate number of dimensions for LSI, K_0 is is obtained.

EXPERIMENTS

Through following two examples and two simulations, we try to explore the performance of the proposed model, in particular, the relationship between $\Phi(K)$ and K.

Experiment 1

Table 1 shows Dataset 1 collected by Deerwester, et al (Deerwester et al. 1990).

No	Documents (Titles)
D1	Human machine interface for Lab ABC computer applications
D2	A survey of user opinion of computer system response time
D3	The EPS user interface management
D4	System and human system engineering testing of EPS
D5	Relation of user-perceived response time to error measurement
D6	The generation of random, binary, unordered trees
D7	The intersection graph of paths in trees
D8	Graph minors IV: Widths of trees and well-quasi-ordering
D9	Graph minors: A survey

Table 1. Dataset 1

Dataset 1 consists of nine titles of Bellcore technical memoranda. Words occurring in more than one document and not on a stop list of 439 common words used by SMART (Sparck 1972) are selected for indexing. They are underlined. Table 2 shows the word-document matrix $A = (a_{mn})_{12\times9}$. The elements of this matrix are the frequencies in which a word occurs in a document. For example, in document D2, the second column of the word-document matrix, *survey*, *user*, *computer*, *system*, *response,* and *time* all occur once. Through running the model (10) with setting an equal value to all w_{ij}, $i = 1,\ldots,N-1$, $j = i+1,\ldots,N$, for the data in Table 2, we have results shown in Figure 1. In Figure1, we find that the value of $\Phi(K)$ goes up as K increases at the beginning. When the value of K is equal to 2, $\Phi(K)$ reaches the maximum value of 27.3. That is $K_0 = 2$. The value of $\Phi(K)$ then goes down while K increases continuously. It means that LSI achieves the best performance in the case of $K_0 = 2$. This result is identical with that in (Deerwester et al. 1990).

Words	Documents								
	D1	D2	D3	D4	D5	D6	D7	D8	D9
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

Table 2. Word-document matrix $A = (a_{mn})_{12\times9}$ for Dataset 1

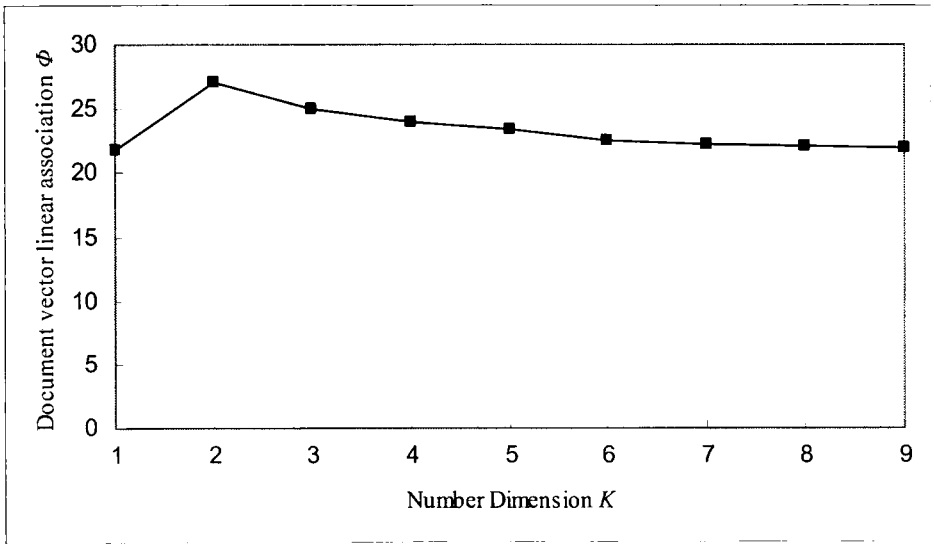

Figure 1. Document vector Linear Association Φ vs. Number Dmension K for Dataset 1

Experiment 2

Table 2 shows dataset 2 collected by Berry et al (Berry et al. 2002). Dataset 2 consists of 15 book titles from Book Reviews published in 1993, Volume 54, No4 of SIAM Review (Berry et al. 2002). We use the same way as Experiment 1 to process words. Table 4 shows the word-document matrix $A = (a_{mn})_{16\times15}$.

No	Documents (Titles)
D1	Automatic differentiation of algorithms: theory, implementation and application
D2	Geometrical aspects of partial differential equations
D3	Ideals, varieties, and algorithms -- an introduction to computational algebraic geometry and commutative algebra
D4	Introduction to hamiltonian dynamical systems and the N-Body problems
D5	Knapsack problems: algorithms and computer implementations
D6	Methods of solving singular systems of ordinary differential equations
D7	Nonlinear systems
D8	Ordinary differential equations
D9	Oscillation theory for neutral differential equations with delay
D10	Oscillation theory of delay differential equations
D11	Pseudodifferential operators and nonlinear partial differential equations
D12	Sinc methods for ouadrature and differential equations
D13	Stability of stochastic differential equations with respect to Semi-Martingales
D14	The Boundary integral approach to Static and Dynamic Contact problems
D15	The double Mellin-Barnes type integrals and their applications to convolution theory

Table 3. Dataset 2

	Documents														
Words	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
algorithms	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
application	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
delay	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
differential	0	1	0	0	0	1	0	1	1	1	1	1	1	0	0
equations	0	1	0	0	0	1	0	1	1	1	1	1	1	0	0
implementation	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
integral	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
introduction	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
methods	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
nonlinear	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
ordinary	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
oscillation	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
partial	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
problem	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0
systems	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
theory	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1

Table 4. Word-document matrix $A = (a_{mn})_{16\times15}$ for Dataset 2

Figure 2 shows the $\Phi(K) \sim K$ curve. It is similar to the result of Experiment 1. The value of $\Phi(K)$ goes up as K increases at the beginning. When the value of $\Phi(K)$ is equal to 3, $\Phi(K)$ reaches the maximum value of 88.127. That is $K_0 = 3$. The value of $\Phi(K)$ then goes down while K increases continuously. It means that the LSI achieves the best performance in the case of $K_0 = 3$. This result is again identical with that in (Berry et al. 2002).

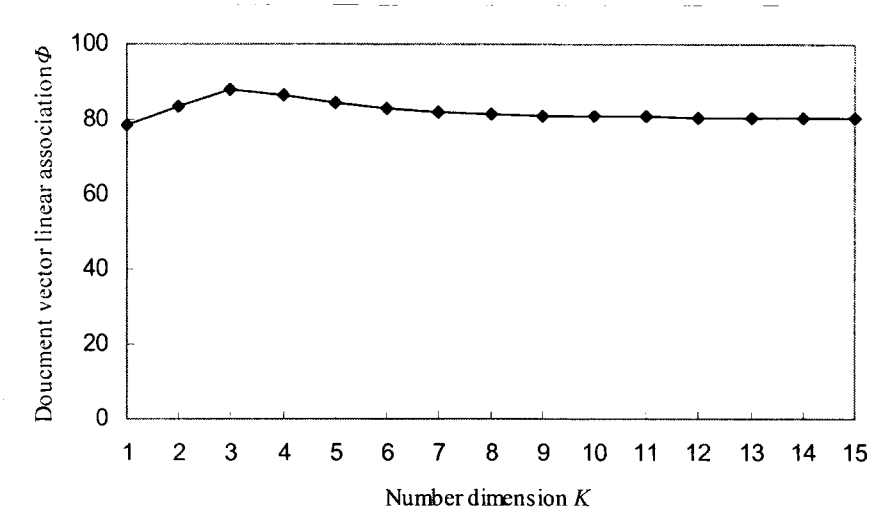

Figure 2. Document Vector Linear Association Φ vs. Number Dimension K for Dataset 2

Simulation Experiment 1

We generated a dataset consisting of 1000 documents by using simulation. There are 100 catalogues. Each catalogue has 10 documents and each document consists of 150 words which are randomly generated within a certain range. For a catalogue C_i $(i = 1,\ldots,100)$, the range is from $((i-1)*5000+1)$ to $i*5000$. Here we assume that one digital number is one word. Words occurring in more than one document are selected for indexing. We have the word-document matrix $A = (a_{mn})_{16974\times1000}$. Figure 3 shows the Φ-K curve.

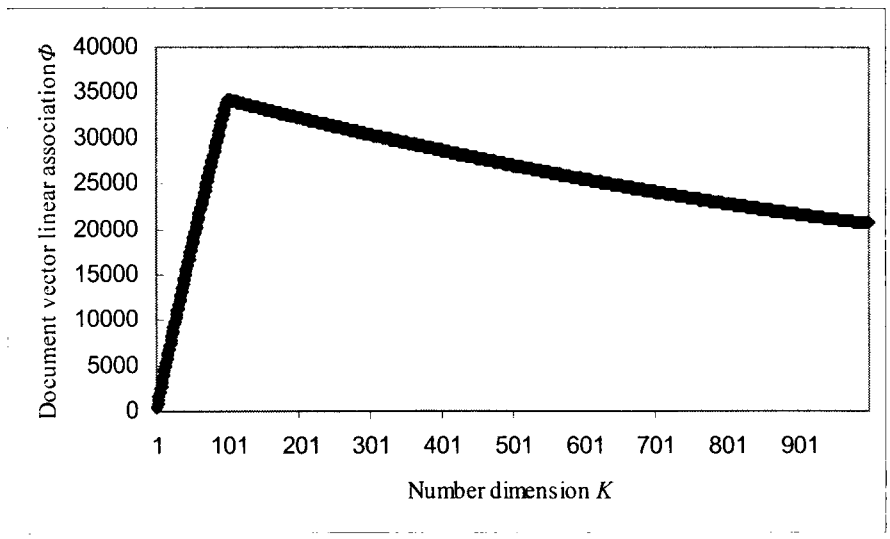

Figure 3. Document vector linear association Φ vs. number dimension K for simulation dataset 1

It is similar to the result of experiment 1 and 2. The value of Φ goes up as K increases at the beginning; when the value of K is equal to 100 (Shown in Figure 4), Φ reaches the maximum value of 34243.5. The value of Φ then goes down while K increases continuously.

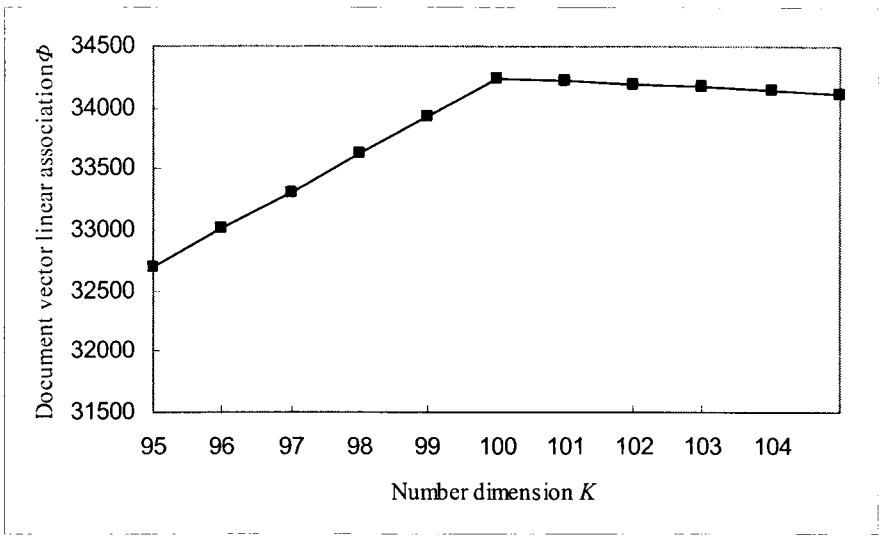

Figure 4. Document Vector Linear Association Φ vs. Number Dimension *K* for Simulation Dataset 1

Simulation Experiment 2

Similarly to the result of simulation experiment 1, we generated a dataset consisting of 1000 documents by using simulation. There are 50 catalogues. Each catalogue has 10 documents and each document consists of 150 words which are randomly generated within a certain range. For a catalogue C_i $(i = 1,\ldots,50)$ the range is from $((i-1)*5000+1)$ to $i*5000$. For the rest 500 documents, each document also consists of 150 words which are randomly generated in the whole range (i.e. from 1 to 100*5000). We use the same way to process words as that of simulation experiment 1. We have the word-document matrix $A = (a_{mn})_{20684\times1000}$. Figure 5 shows the Φ-$K$ curve. It is similar to the result of simulation experiment 1. The value of Φ goes up as K increases at the beginning; when the value of K is equal to 50 (Shown in Figure 6), Φ reaches the maximum value of 43797.95. The value of Φ then goes down while K increases continuously.

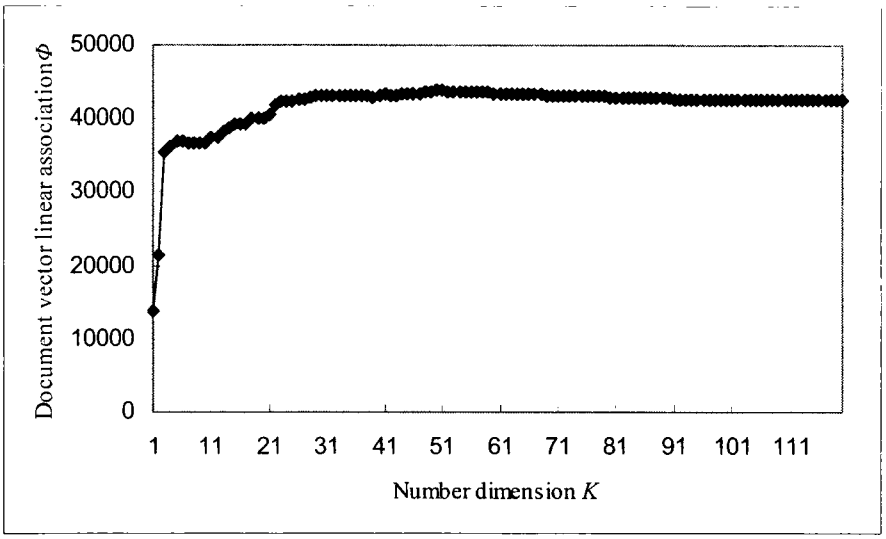

Figure 5. Document Vector Linear Association Φ vs. Number Dimension *K* for Simulation Dataset 2

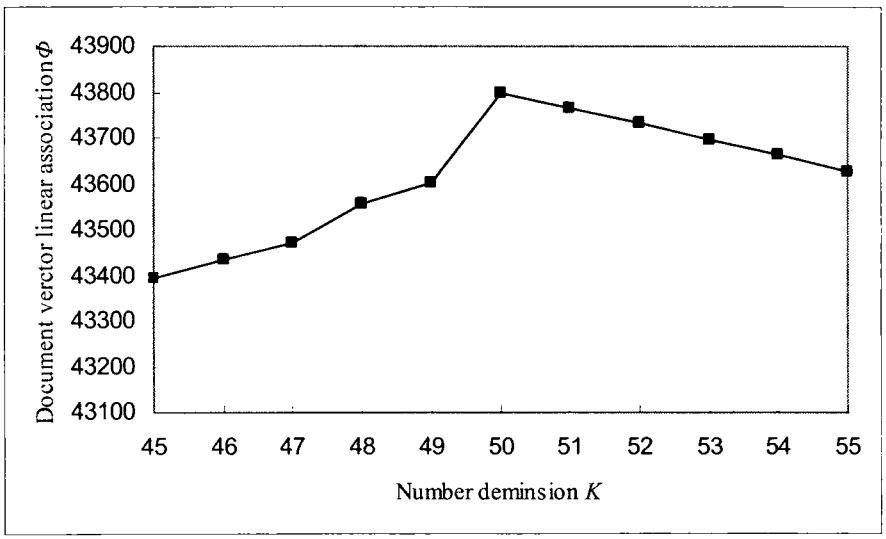

Figure 6. Document Vector Linear Association Φ vs. Number Dimension *K* for Simulation Dataset 2

CONCLUSIONS AND FUTURE WORK

Choosing an appropriate number of dimensions for the LSI is a principle problem for the application of LSI technology. This presents a document vector model to solve this problem by using weighting technology. The experimental results demonstrate that this method can determine an appropriate number of dimensions for LSI. Further investigations will be carried out to apply this method to large datasets. We will explore its potential ability to detect a real dataset structure.

REFERENCES

1. Bellegarda, J. R. (1998) Exploiting both local and global constraints for multi-span statistical language modeling, http://www.telecom.tuc.gr/paperdb/icassp98/pdf/scan/ic981164.pdf.
2. Berry, M. W., Dumais, S. T. and Obrien, G. W (2002) Using Linear Algebra for Intelligent Information Retrieval, http://citeseer.nj.nec.com/rd/37422678,19079,1,0.25,Download/http://citeseer.nj.nec.com/cache/papers.
3. Cristianini, N., Shawe, J. and Lodhi, H. (2002) Latent Semantic Kernels, *Journal of Intelligent Information Systems*, 18, 2/3, 127–12.
4. Cullum, J. K. and Willoughby, R. A, (1985) Lanczos algorithms for large symmetric eigenvalue computations, Brikhauser, Boston.
5. Deerwester, S., Dumais, S., Furnas, T. G. W. Landauer, T. K., and Harshman, R. (1990) Indexing by latent semantic analysis, *Journal of the American Society for Information Science*, 41, 4, 391–407.
6. Dumais, S. T. (1998) LSI meets TREC: A status report, http://trec.nist.gov/pubs/trec2/papers/txt/10.txt.
7. Dumais, S. T. (1991) Improving the retrieval of information from external sources, *Behavior Research Methods, Instruments and Computers*, 23, 2, 229–236.
8. Foltz, P.W. and Dumais, S. T. (1992) An analysis of information filtering methods, *Communications of the ACM*, 35, 12, 51–60.
9. Gene, H., Franklin T., and Michael L. (1981) A block lanczos method for computing the singular values and vorresponding singular vectors of a matrix, *ACM Transactions on Mathematical Software*, 17, 2, 149–169.
10. Kawamae, N. (2002) Latent semantic indexing based on factor analyssis, http://ultimavi.arc.net.my/banana/Workshop/SCI2002/papers/Kawamae.pdf.
11. Landauer, T. K. and Dumais, S. T. (1997) A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge, *Psychological Review,* 104, 2, 211–240.
12. Landauer, T. K., Laham, D., Rehder, R., and Schreiner, M. E. (1997) How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and humans, http://lsa.colorado.edu/papers/cogsci97.pdf.
13. Rehder, B., Schreiner, M., Laham, D., Wolfe, M., Lan-dauer, T., & Kintsch, W., (1998) Using LatentSemantic Analysis to assess knowledge: Some tech-nical considerations, *Discourse Processes*, 25, 337–354.
14. Sakawa, T. (1993) Interactive multiobjective linear programming with fuzzy parameters, Fussy *sets and interactive mulitobjective optimization*. New York: Plenum Press.
15. Salton, G. and McGill, M. J. (1983) Introduction to Modern Information Retrieva, McGraw–Hill.
16. Sparck, J. K. (1972) A statistical interpretation of term specificity and its applications in retrival, *Journal of Documental*, 28, 1, 11–21.
17. Wiemer, P. el at. (1999) Improving an intelligent tutor's compre-hension of students with Latent Semantic Analysis, In Lajoie, S., & Vivet, M. (Eds.), *Artificial Intelligence in Education*, (Amsterdam, IOSPress), 535–542.
18. Wiemer, P., (2000) Adding syntactic information to LSA, http://reed.cs.depaul.edu/peterwh/papers/cogsci00.pdf.