Journal of Computing and Information Technology - CIT 14, 2006, 3, 191–196
doi:10.2498/cit.2006.03.02

Latent Semantic Indexing-Based Intelligent Information Retrieval System for Digital Libraries

Ch. Aswani Kumar, Ankush Gupta, Mahmooda Batool and Shagun Trehan

School of Computer Sciences, Vellore Institute of Technology, Deemed University, India.

To the information retrieval research community, a digital library can be viewed as an extended information retrieval system. The primary goal of an information retrieval system is to retrieve all the relevant documents, which are relevant to the user query. Disparities between the vocabulary of the system's authors and that of their users pose difficulties when information is processed without human intervention. In this paper, we present a novel approach to enhance the efficiency of the information retrieval system using intelligent information processing technique. Experiments carried out are giving most encouraging results.

Keywords: digital libraries, information retrieval, latent semantic indexing, singular value decomposition, vector space model.

1. Introduction

The growing number of electronic textual documents has created the need of intelligent access to the information implied by them. The unstructured nature of documents, however, makes it difficult to realize the goal. In the present scenario a large volume of data is available related to each and every issue. This increase in the amount of data has led to a situation where users are swamped with information and have difficulty sifting through the reams of material, much of which is not relevant to them. This is commonly referred to as the problem of information overload [1][2].

A typical Information Retrieval (IR) system responds to the user's query by selecting documents and ranking them in terms of relevance. To be effective in its attempt to satisfy the user information need, an information retrieval system must somehow interpret the content of the information items in a collection and rank them according to the degree of their relevance to the user query. This interpretation of document content involves extracting the syntactic and semantic information from the document text and using this information to match user information need.

The implementation of Digital Library (DL) as an information service center is an interesting research area that has grown out of developments in both network and multimedia technology. DLs require research in many areas including dynamic interoperability support for library evolution, contextual search mechanisms and social issues. Such research requires interdisciplinary efforts from information systems, computer science, library science, management science, social science and other fields. IR is essential for the success of DLs, so they can achieve high levels of effectiveness while at the same time affording ease of use. People's needs still leave a rich research agenda for the IR community. Now there are DLs at universities, publishers, government agencies and public libraries [3][4]. It is appropriate to see how IR may expand its horizons to deal with the key problems of DLs and how it can provide a unifying and integrated framework for digital library field. There is a need for sophisticated IR systems with intelligent processing techniques.

This paper is structured as follows: Section 2 presents the Vector Space Model. Latent Semantic Indexing is described in Section 3. Section 4 gives the details of intelligent processing approach and implementation. Section 5 presents the results of document collection and it is followed by conclusions and references.

2. Vector Space Method (VSM)

The Vector Space Model (VSM) models the documents as a set of terms that can be individually weighted and manipulated; performs queries by comparing the representation of a query to the representation of each document in the space and retrieving relevant documents [1][5]. In the VSM, each document $\vec{d_j}$ is represented by a weight vector $\vec{d_j} = (w_{1j}, w_{2j}, \text{- - -}, w_{tj})^T$, where w_{zj} is the weight or importance of the term z in representation of the document $\vec{d_j}$, t is the size of the indexing term set. A collection of n documents is then represented by $a\,t\,x\,n$ term by document matrix.

In VSM, two main components of the term *weight* are used to represent the elements of the term by document matrix, the frequency (TF) of occurrences of each word in a particular document and the inverse document frequency (IDF) of each word, which varies inversely with the number of documents to which a word is assigned. So, the weight of a term i in a document j is given by the following equation:

$$w_{i,j} = tf_{i,j} * idf_i = tf_{i,j} * \log(N/df_i) \qquad (1)$$

where $tf_{i,j}$ is the frequency of the i^{th} term in j^{th} document, df_i is number of documents in which term i appears at least once, N is the number of documents in the collection. This method assigns the highest weight to those terms which appear frequently in a small number of documents in the documents set.

For queries, the same vector representation is given as $\vec{q}_i = (q_{1i},\ q_{2i}, \text{- - -}, q_{ti})^T$, q_{zi} is the weight of term z in representation of the query $\vec{q_i}$. We measure the similarity between a document and a query and both are normalized to unit length, in underlying vector space,

$$\text{Sim}_{\text{VSM}}\ \vec{q_i}, \vec{d_j} = \frac{\sum_{z=1}^{t} (q_{zi} w_{zj})}{\sqrt{\sum_{z=1}^{t} q_{zi}^2} \sqrt{\sum_{z=1}^{t} w_{zi}^2}} \qquad (2)$$

The advantages of this approach are adaptability, robustness and minimal user intervention.

3. Latent Semantic Indexing (LSI)

In VSM, since normally not every word appears in each document, the document matrix is usually of a high dimension and sparse. High dimensional and sparse matrices are susceptible to noise and have difficulty in capturing the underlying semantic structure. In addition, storage and processing of such data places great demands on computing resources [6][7].

Reduction of the dimensionality of the model is one way to address this problem. Singular Value Decomposition (SVD) takes advantage of the implicit higher order structure in the association of terms within documents by largest singular vectors. The vectors representing the documents are projected in new, low dimensional space obtained by SVD [8][9][10][11]. The dimensionality reduction is accomplished by approximating the original term-by-document A with a new matrix A_k where rank $(A) = r >$ rank $(A_k) = k$. In SVD, a large term by document matrix is decomposed into a set of orthogonal factors from which the original matrix can be approximated by linear combination. Vectors of factor weights represent documents. Queries are represented as pseudo-document vectors formed from weighted combinations of terms, and documents with supra-threshold cosine values are returned. The SVD of matrix A is written as

$$A = U\,\Sigma\,V^T \qquad (3)$$

If the term-by-document matrix A is $t \times d$, then U is a $t \times d$ orthogonal matrix, V is a $d \times d$ orthogonal matrix; Σ is a $d \times d$ diagonal matrix where the values on the diagonal of Σ are called the Singular Values. By changing all but the top k rows of Σ to zero rows, a low rank approximation to A called A_k can be created,

$$A_k = U_k \Sigma_k V_k^T \qquad (4)$$

Where U_k is the $t \times k$ term-by-concept matrix, Σ_k is $k \times k$ concept-by-concept matrix; V_k is $k \times d$ concept-by-document matrix. The rank of A has been lowered from r to k. This low rank approximation removes redundancy from original data and allows us to uncover latent semantic relations among terms as well as documents.

Queries are formed into pseudo-documents that specify the location of the query in the reduced term-document space. Given a query vector the pseudo-document, $q^{\wedge}$, can be represented by

$$q^{\wedge} = q^T U_k \Sigma_k^{-1} \qquad (5)$$

Thus, the pseudo-document consists of the sum of the term vectors $(q^T U_k)$ corresponding to the terms specified in the query scaled by the inverse of singular values (Σ_k^{-1}). Once the query is projected into the term-document space, one of several similarity measures can be applied to compare the position of the pseudo-document with the positions of terms or documents in the reduced term-document space.

4. An Intelligent Information Retrieval based on LSI

It is very important for DLs to provide the information that users wanted. In a conventional library, librarians improve the reliability of reference services by solving the problems of omission or noise caused by differences in requests among individual users. In a DL, interaction between users and librarians decreases. A DL should improve the reliability of its services by automatic means. Hence there is a need for intelligent techniques to bridge the semantic gap between users requests and DLs. In this section, we present an intelligent agent that preprocesses the information.

Deciding about the importance of a term for summarizing the content of a document is not a trivial issue. Despite this difficulty, there are properties of an index term, which are easily measured and useful for evaluating the potential of a term as such. Thus, it should be clear that distinct index terms have varying relevance when used to describe the document contents. This effect is captured through the assignment of numerical weights to each index term of a document.

4.1. TF- IDF Normalization

Content of a document is determined by relative frequencies of the term and not by the total number of times particular terms appear in the document. So we scaled our document collection using the weighting model TF $\times$ IDF (Term Frequency $\times$ Inverse Document Frequency).

4.2. Stemming Algorithm

In constructing a term-by-document matrix, terms are usually identified by their word stems. Stemming reduces the storage requirements by decreasing the number of words maintained. The stemming algorithm normally used is Porter Stemming Algorithm [5]. The Porter stemming algorithm removes the commoner and morphological endings from English words. It doesn't usually matter whether the stems generated are genuine words or not – thus, "computation" might be stemmed to "comput". We have appended an intelligent processing module to the existing Porter Stemming algorithm.

4.3. Working Principle of Intelligent Processing Approach

Step 1. After the user submits the query, the query is tokenized. Generated tokens of the query are compared with the elements of the array containing the keywords generated previously. Taking each and every token one by one, if the token matches any keyword, it is represented by 1. If the token does not match any of the keywords then it is sent to step two.

Step 2. Check whether the token ends with 's'.

a) If yes, then remove 's' from the token and compare it with the keywords. If it matches with any of the keywords then 1 is placed, else we concatenate 's' to the token and send it to the next step.

b) If no, then the token is sent to the next step.

Step 3. In this step, the token is sent to stemming algorithm and the stemmed token is matched with the keywords.

If the token matches with any of the keywords, then it is represented by 1 else by 0.

The steps through 2 to 3 are enhancements over the original method of Porter Stemming for query formulation.

4.4. Design of the System

The system design is shown in Figure 1. When the user submits the query to the system, initially it will be pre-processed by the intelligent query-processing agent. This intelligent agent [12] processes the query based on the approach explained in the section 4.3. The processed query will be matched against the document collection to calculate the cosine similarity between the query vector and document vectors

Fig. 1. Design of the System.

as shown in Figure 2. Then the resulting documents will be ranked based on the cosine similarity. As per ranking order, documents will be displayed to the user as shown in Figure 3.

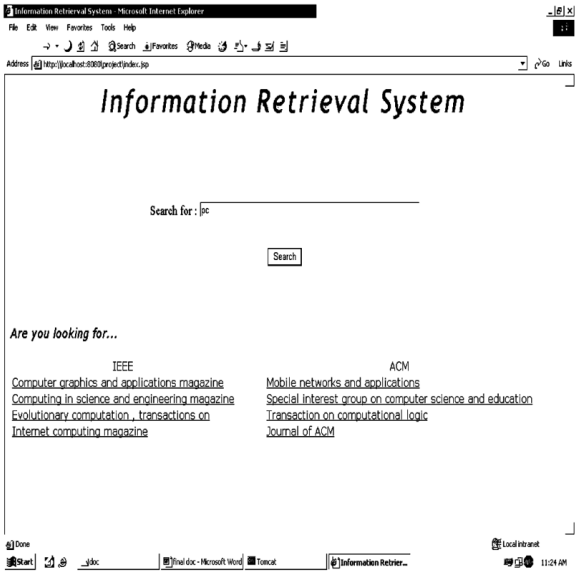

Fig. 2. Querying the Document Collection.

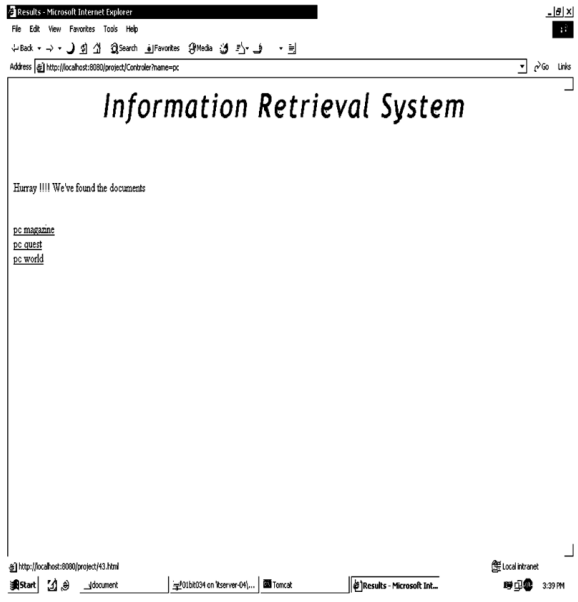

Fig. 3. Page Displaying Results.

5. Evaluation and Result Analysis

Recall and Precision are two metrics to evaluate the efficiency of the IR system. Recall refers to how successful the system is in retrieving all the documents that possibly relate to a search query. Precision refers to how well the system separates irrelevant documents from the truly relevant ones.

Experiments were carried out to evaluate the effectiveness of the proposed system. With the help of domain expert we have generated a term by document matrix of dimension 153×116 from a corpus containing titles of 116 journals and periodicals available in the central library of Vellore Institute of Technology. Due to the space constraints, we could present only a sample list of documents from the corpus, along with their word stems in Table 1. The original rank of the term-by-document matrix of the corpus is 116. The matrix is reduced to the approximated rank 105. After making a

No	Title of the Journal	Word Stems
1.	Annals of the history of the computing – IEEE	annals, history, computing, ieee.
2.	Information theory, transactions on – IEEE	information, theory, transactions, ieee.
3.	Journal of computer documentation – ACM	journal, computer, documentation, acm.
4.	Knowledge and data engineering, transactions on – IEEE	knowledge, data, engineering, transactions, ieee.
6.	DESIDOC Bulletin of Information Technology	desidoc, bulletin, information, technology.
7.	IE Journal Of Computer Engineering Division	ie, journal, computer, engineering, division.

Table 1. Sample of the Data used for Evaluation.

few iterations, we identified that rank of 105 gives better results for the implemented document collection. How to choose the rank that provides optimal performance of LSI for any given document collection still remains an open question. However, the optimal rank is collection dependent. We have formulated a set of queries to match against the approximated term-by-document matrix. A common way to evaluate the performance of retrieval methods is to compute interpolated precision at various standard recall levels.

Interpolated precision values for standard recall levels for VSM and LSI using both traditional and intelligent approaches is given in Table 2. The values show the superiority of intelligent approach over traditional approach for both VSM and LSI.

Methods	**Average Precision**
LSI with Traditional Approach	59.06
LSI with Intelligent Approach	**60.99**
VSM with Traditional Approach	44.85
VSM with Intelligent Approach	**59.79**

Table 2. Interpolated Precision Values for Different Methods.

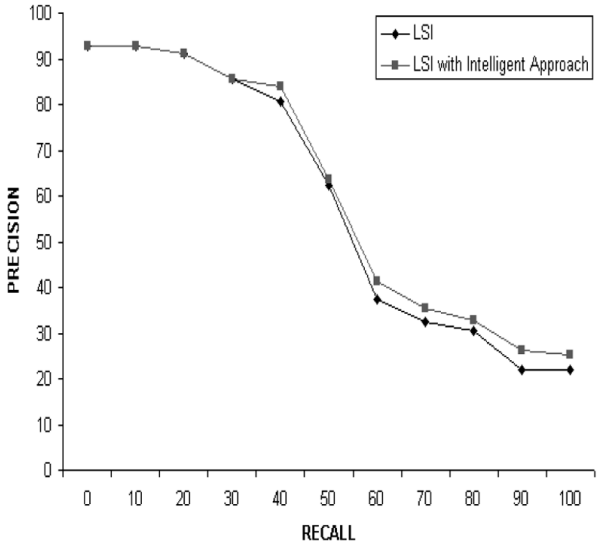

Fig. 4. LSI with Tradicional and Intelligent Approach.

Figure 4 shows the superiority of LSI using intelligent approach over LSI using traditional approach. It yielded better precision results at the Recall levels above 30. Exhibiting superior results at higher recall levels is the most significant advantage.

Figure 5 presents the consolidated results for VSM and LSI using traditional and intelligent approaches. For all standard recall levels, precision of LSI is higher than that of VSM. Using the intelligent approach for the recall levels up to 60, LSI and VSM have given same precision values, but for higher Recall levels, above 60, LSI has shown its superiority.

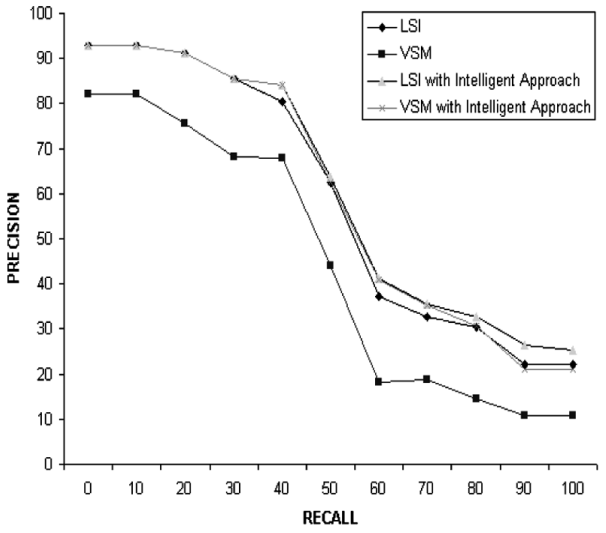

Fig. 5. Copmparison of VSM and LSI using Tradicional and Intelligent methods.

6. Conclusion

In this paper, we introduced a novel intelligent approach to enhance the efficiency of IR system. The experiments that were conducted gave most promising results showing the superiority of our approach over traditional methods with VSM and LSI. The system can be deployed in information intensive applications such as digital libraries. In future, we would like to experiment with other variants of LSI to investigate performance of the IR system.

Acknowledgements

Authors acknowledge the financial support from Department of Science and Technology, Govt. of India under the grant number SR/S3/EECE/25/2005.

References

[1] R.B. YATES, B.R. NETO, Modern Information Retrieval. *Pearson Education*, 1999.

[2] N.J. BELKIN, W.B. CROFT, Information Filtering and Information Retrieval: Two sides of the same coin?. *Communications of the ACM*, 35(1992), 29–38.

[3] CH. ASWANI KUMAR, S. SRINIVAS, On Adopting Software Agents for Distributed Digital Libraries. *DESIDOC Bulletin of Information Technology*, 24(2004), 3–8.

[4] YUEYU FU, JAVED MOSTAFA, Toward Information Retrieval Web Services for Digital Libraries. *Proc. of IEEE/ACM Joint Conference on Digital Libraries*, 2004.

[5] MICHAEL W. BERRY, ZLATKO DRMAC, Matrices, Vector Spaces, and Information Retrieval. *SIAM Review*, 41(1999), 335–362.

[6] SCOTT DEERWESTER, Indexing by latent semantic analysis, Journal of the American Society for Information Science, 41(1990), 391–407.

[7] M.W. BERRY, S.T. DUMASIS, Using Linear Algebra for Intelligent Information Retrieval, *SIAM Review*, 37(1995), 573–995.

[8] A. KONTOSTATHIS, W.M. POTTENGER, A Mathematical View of Latent Semantic Indexing: Tracing Term Co occurrences, *Lehigh University Technical Report*, LU-CSE-02-006, 2002.

[9] JING CAO, JUN ZHANG, Clustered SVD Strategies in Latent Semantic Indexing Information Processing and Management, 2005 (Article in Press).

[10] K. APRIL, WILLIAM M. POTTENGER, " A Framework for Understanding Latent Semantic Indexing Performance" , *Journal of Information Processing and Management*, 2005. (Article in Press).

[11] HOLGER BAST, INGMAR WEBER, " Insights from Viewing Ranked Retrieval as Rank Aggregation", *Proc. of Workshop on Challenges in Web Information Retrieval and Integration*, WIRI05, 2005.

[12] CH. ASWANI KUMAR, S. SRINIVAS, Agent-based approach for Distributed Information Retrieval, *ACCST Research Journal*, 2(2004), 6–9.

Received: August, 2005
Revised: February, 2006
Accepted: February, 2006

Contact addresses:
Ch. AswaniKumar
Lecturer, School of Computer Sciences,
Vellore Institute of Technology
Deemed University
Vellore - 632014, India.
e-mail: aswani@vit.ac.in.

Ankush Gupta
School of Computer Sciences
Vellore Institute of Technology
Deemed University
Vellore - 632014, India.

Mahmooda Batool
School of Computer Sciences
Vellore Institute of Technology
Deemed University
Vellore - 632014, India.

Shagun Trehan
School of Computer Sciences
Vellore Institute of Technology
Deemed University
Vellore - 632014, India.

CH. ASWANI KUMAR, is Lecturer at School of Computer Sciences, Vellore Institute of Technology, Vellore –632014, India. His areas of interest include Information Retrieval, Machine Intelligence.

ANKUSH GUPTA, received his B. Tech degree in Information Technology from Vellore Institute of Technology, India. Currently he is project associate in Accenture Technologies, Bangalore, India. His research interests include Information Retrieval, Data Mining.

MAHMOODA BATOOL, received her B.Tech degree in Information Technology from Vellore Institute of Technology, India. Currently she is project trainee in Cognizant Technologies, Chennai, India. Her research interests include Data Mining, Database Systems.

SHAGUN TREHAN, received her B.Tech degree in Information Technology from Vellore Institute of Technology, India. Currently she is project trainee in Cognizant Technologies, Pune, India. Her research interests include Data Mining, Database Systems.