自然言語処理 133−10
(1999. 9. 10)

離散フーリエ変換を用いたベクトル空間モデルの次元削減

北 研二　佐々木 稔

徳島大学 工学部
〒770-8506　徳島市南常三島町 2-1
kita@is.tokushima-u.ac.jp

概要

代表的な情報検索モデルであるベクトル空間モデルの次元を離散フーリエ変換により削減する方法を提案する。また、提案した方法を概念に基づく検索モデルである潜在的意味インデキシング法 (Latent Semantic Indexing; LSI) へ適用することを試みる。

Dimensionality Reduction of Vector Space Model based on Discrete Fourier Transform

Kenji Kita　Minoru Sasaki

Faculty of Engineering, Tokushima University
Tokushima 770-8506, Japan
kita@is.tokushima-u.ac.jp

Abstract

In this paper, we propose to use the Discrete Fourier Transform (DFT) for dimensionality reduction of the vector space information retrieval model. The point is to apply DFT to document vectors and use the first several Fourier coefficients as document features. We also apply DFT-based dimensionality reduction to Latent Semantic Indexing (LSI). Instead of performing the Singular Value Decomposition (SVD) on the entire term-document matrix, we show that it is sufficient to perform SVD on a DFT-derived reduced space.

1 Introduction

The Vector Space Model (VSM) is a conventional information retrieval model, which represents documents and queries by vectors in a multidimensional space. The basic idea is to extract indexing terms from a document collection and to represent each document or query as a vector of weighted term frequencies. The similarity comparison among documents, and between documents and queries, is performed via the similarity between two vectors (e.g. cosine similarity). In VSM, the document vectors exhibit the following two properties:

1. Since the size of the indexing terms is typically large, the document vectors are high-dimensional.

2. Since the number of terms in one document is typically far less than the total number of indexing terms, the document vectors are very sparse.

High-dimensional and sparse vectors are susceptible to noise, and are difficult to capture the underlying structure. Additionally, the storage and processing of such data places great demands on computing resources. Dimensionality reduction is a way to overcome these problems. It maps vectors in a high-dimensional space to vectors in a lower dimensional feature space. A variety of dimensionality reduction techniques have been studied extensively in statistical pattern recognition and matrix algebra, such as Singular Value Decomposition (SVD) [12], Karhunen-Loève Transform or FastMap [9]. These techniques can be fine-tuned to the specific data set, and therefore they can achieve good performance. However, they do not work adaptively and so are often much too slow for real-time application.

In a context of information retrieval, Latent Semantic Indexing (LSI) [2, 4] uses truncated SVD to reduce the dimensionality of the term-document space. LSI has demonstrated improved performance over the conventional VSM for several document collections [4, 7, 13]. Despite its effectiveness, however, SVD is computationally expensive for a large document collection. Researchers have tackled this problem using a variety of approximation techniques [5, 11, 14, 15, 17].

In the work described here, we use the Discrete Fourier Transform (DFT) for dimensionality reduction of the vector space model. DFT has been successfully used for indexing time-series databases for similarity searching [1, 8]. Here, we apply the same technique to the vector space information retrieval model. We also show that DFT-based dimensionality reduction yields an improvement on LSI by using a two-step method: we first apply DFT to the original term-document matrix, and then we perform the SVD on a DFT-derived reduced space.

2 Discrete Fourier Transform

We first briefly review the Discrete Fourier Transform (DFT) [16]. The N-point discrete Fourier transform of the sequence x_n $(n = 0, \cdots, N-1)$ is defined as:

$$X_k = \sum_{n=0}^{N-1} x_n \, e^{-j\frac{2\pi}{N}nk} \quad (k = 0, \cdots, N-1) \tag{1}$$

where j is the imaginary unit. The sequence x_n can be recovered from X_k by the inverse discrete Fourier transform as follows:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{j\frac{2\pi}{N}nk} \quad (n = 0, \cdots, N-1) \tag{2}$$

The discrete Fourier transform can be computed in $O(N \log N)$ time with an algorithm called the Fast Fourier Transform (FFT).

The following theorem holds, which is known as Parseval's theorem:

$$\sum_{n=0}^{N-1} |x_n|^2 = \sum_{k=0}^{N-1} |X_k|^2 \tag{3}$$

Since the DFT is linear transformation, Parseval's theorem implies that the Euclidean distance between two sequences x and y is preserved:

$$||x - y||^2 = ||X - Y||^2 \tag{4}$$

where X and Y are DFTs of x and y, respectively.

3 FFT-based Vector Space Model

We now describe our vector space information retrieval model which incorporates FFT-based dimensionality reduction. The point is to apply FFT to m-dimensional document vectors and use the first m' Fourier coefficients ($m' < m$) as document features, dropping all other Fourier coefficients. Using DFT for dimensionality reduction or feature extraction seems promising because, as Agrawal et al. (1993) pointed out, for a large number of sequences of practical interest, only the first few frequencies are strong.

The following is a resume of the FFT-based vector space model:

1. Extract indexing terms from the entire document collection using an appropriate stop list and stemming algorithm. Let we have m indexing terms and n documents.

2. Create n document vectors $d_1, d_2, \cdots, d_n$ where the i-th component of document vector d_j is defined as follows:

$$a_{ij} = L_{ij} \times G_i \tag{5}$$

   Here, L_{ij} is the local weighting for the i-th term in document d_j, and G_i is the global weighting for the i-th term.

3. Create normalized document vector x_i so that each document vector has unit norm:

$$x_i = \frac{1}{||d_i||} d_i \quad (i = 1, \cdots, n) \tag{6}$$

4. Apply the fast Fourier transform to each normalized document vector x_i.

5. Build a multidimensional index using the first m' Fourier coefficients, where m' stands for cut-off frequency. Thus, each document is represented as a point in a m'-dimensional space.

6. For purposes of information retrieval, use the same transformation to map a query vector into a m'-dimensional space.

The similarity between documents and queries is typically measured by the cosine similarity of their vectors. The method above normalizes the document and query vectors to be points on the unit hyper-sphere. Furthermore, all components of the document and query vectors are non-negative, hence the following relationship holds:

$$\cos(q, x_i) > \cos(q, x_j) \iff ||q - x_i||^2 < ||q - x_j||^2 \tag{7}$$

where q is a normalized query vector and x_i, x_j are normalized document vectors. Since DFT preserves the Euclidean distance, retrieval can be performed on the frequency domain.

The greatest advantage of the FFT-based dimensionality reduction is that it works directly with the document data. That is, it can be implemented adaptively so the IR model is updated after a new document is given, without need of reusing all the document data.

4 FFT-based Latent Semantic Indexing

The FFT-based technique can be used to reduce the dimension of the document space, but it does not bring together semantically related documents. Latent Semantic Indexing (LSI) is an extension of VSM which addresses the latter problem.

4.1 Latent Semantic Indexing

LSI uses the Singular Value Decomposition (SVD) to factor a term-document matrix $A = [a_{ij}] \in \mathcal{R}^{m \times n}$ into the product of three matrices:

$$A = U \Sigma V^T, \quad r = \text{rank}(A) \tag{8}$$

where $U \in \mathcal{R}^{m \times r}$ and $V \in \mathcal{R}^{n \times r}$ are orthogonal matrices and $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_r)$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, is a diagonal matrix whose diagonal entries are called the singular values of A.

The reduced-dimension representation is then given by the rank-k approximation:

$$A_k = U_k \Sigma_k V_k^T \quad (k < r) \tag{9}$$

where U_k and V_k are formed by the first k columns of U and V respectively, and Σ_k is the k-th leading principal submatrix of Σ. The Eckart-Young theorem states that A_k is the best rank-k approximation of A, namely:

$$\min_{\text{rank}(B)=k} ||A - B||_F = ||A - A_k||_F \tag{10}$$

where $|| \cdot ||_F$ indicates the Frobenius norm:

$$||A||_F^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2 \tag{11}$$

In LSI, each document is projected into a lower dimensional space $\Sigma_k^{-1} U_k^T A$. Queries are also projected into the same space, thus the relevance of documents to a query is $A^T U_k \Sigma_k^{-2} U_k^T q$.

4.2 LSI by FFT-based Dimensionality Reduction

One bottleneck in LSI is its computational cost. SVD computations take time polynomial in m, n which is often too prohibitive for large text collections. FFT-based dimensionality reduction yields an improvement on SVD computations as follows:

1. For the term-document matrix A, normalize its column (document) vectors.
2. Apply FFT-based dimensionality reduction to each column of A to obtain an $m' \times n$ matrix A' $(m > m' > k)$.
3. Apply SVD to A' and use the rank-k approximation A'_k as a reduced term-document space.

5 Experimental Results

An experimental comparison was made among four information retrieval models:

(1) conventional vector space model (VSM),

(2) FFT-based vector space model (FFT-VSM),

(3) latent semantic indexing (LSI),

(4) FFT-based latent semantic indexing (FFT-LSI).

As for FFT-LSI, we chose 1000 and 2000 as dimensions of an FFT-derived reduced space. Hereafter, these models are denoted as FFT1000-LSI and FFT2000-LSI, respectively.

In the experiments, we used the MEDLINE collection, where queries and relevance judgements were available (1033 documents and 30 queries). We first preprocessed documents to eliminate non-content-bearing stopwords using a stop list of 439 common English words. Terms occuring in only one document were also removed. The remaining terms were then stemmed using the Porter algorithm [10]. The preprocessing step resulted in 4329 indexing terms.

As a term weighting scheme, we used Log-Entropy defined as follows [3, 6]:

$$L_{ij} = 1 + \log f_{ij} \tag{12}$$

$$G_i = 1 - \sum_{j=1}^{n} \frac{\frac{f_{ij}}{F_i} \log \frac{f_{ij}}{F_i}}{\log n} \tag{13}$$

where n is the number of documents in the collection, f_{ij} is the frequency of the i-th term in the j-th document, and F_i is the frequency of the i-th term throughout the entire document collection.

For our retrieval evaluation, we measured the non-interpolated average precision, which refers to an average of precision at various points of recall, varying the number of dimensions, using the top 50 documents retrieved. Figure 1 gives average precision results as a function of model's dimensions. For comparison, we included the VSM's result in the figure, though the dimension of VSM is constant (equals the number of indexing terms).

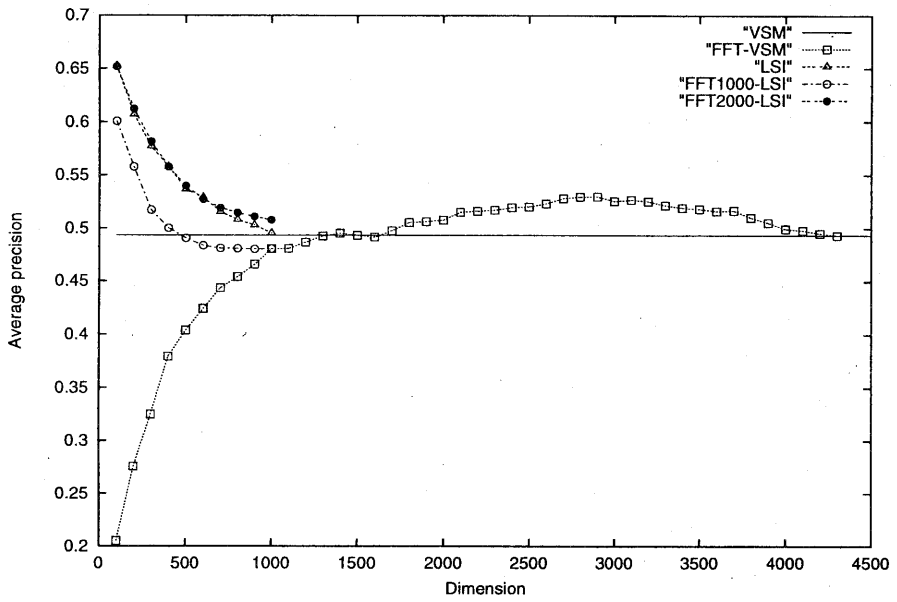

Figure 1: Average precision as a function of dimensions.

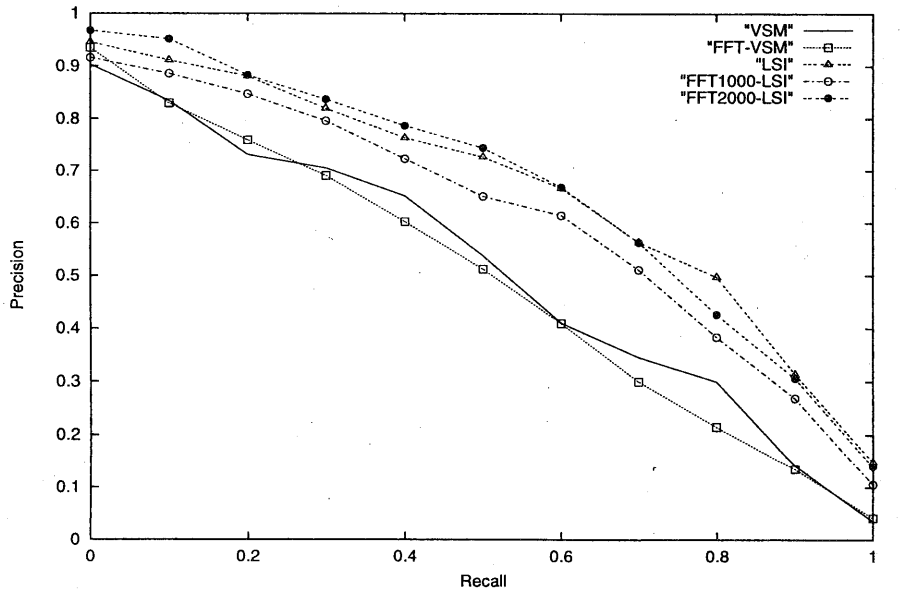

Figure 2: Recall-precision curve.

We also evaluated retrieval models by measuring precision at several different levels of recall using the following models: VSM, 1000-dimensional FFT-VSM, 100-dimensional LSI, 100-dimensional FFT1000-LSI and FFT2000-LSI. Figure 2 gives recall-precision curves, which show interpolated precision as a function of recall.

The results given in Figure 1 and Figure 2 show that the 1000-dimensional FFT-VSM achieved as good performance as 4329-dimensional VSM. 100-dimensional FFT1000-LSI is slightly worse than 100-dimensional LSI, but FFT2000-LSI's performance is comparable to LSI's.

6 Conclusion

We have proposed a method for dimensionality reduction of the vector space information retrieval model using the fast discrete Fourier transform. The proposed method works directly with the document data, and thus it is particularly useful in applications that require frequent document updates. Experimentally, we showed that the proposed method offers improvemnt over the conventional VSM.

We have also showed that FFT-based dimensionality reduction can be applied to latent semantic indexing, by performing the SVD not on the entire term-document matrix but on an FFT-derived reduced space, without great loss of accuracy.

References

[1] Agrawal, R., Faloutsos, C. and Swami, A.: "Efficient similarity search in sequence databases", *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, pp. 69-84, 1993.

[2] Berry, M. W., Dumais, S. T. and O'Brien, G. W.: "Using linear algebra for intelligent information retrieval", *SIAM Review*, 37(4), pp. 573-595, 1995.

[3] Chisholm, E. and Kolda, T. G.: "New term weighting formulas for the vector space method in information retrieval", Technical Memorandum ORNL-13756, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 1998.

[4] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K. and Harshman, R.: "Indexing by latent semantic analysis", *Journal of the American Society for Information Science*, 41(6), pp. 391-407, 1990.

[5] Drineas, P., Frieze, A., Kannan, R., Vempala, S. and Vinay, V.: "Clustering in large graphs and matrices", *Proceedings of the 10th ACM-SIAM Symposium on Discrete Algorithms*, 1999.

[6] Dumais, S. T.: "Improving the retrieval of information from external sources", *Behavior Research Methods, Instruments and Computers*, 23(2), pp. 229-236, 1991.

[7] Dumais, S. T.: "Using LSI for information filtering: TREC-3 experiments", *Overview of the Third Text REtrieval Conference*, Harman, D. K. (ed.), NIST Special Publication 500-226, pp. 219-230, 1995.

[8] Faloutsos, C., Ranganathan, M. and Manolopoulos, Y.: "Fast subsequence matching in time-series databases", *Proceedings of the ACM SIGMOD Conference on Management of Data*, pp. 419-429, 1994.

[9] Faloutsos, C. and Lin, K-I.: "FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets", *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, pp. 163-174, 1995.

[10] Frakes, W. B. and Baeza-Yates, R.: "*Information Retrieval: Data Structures and Algorithms*", Prentice Hall, 1992.

[11] Frieze, A., Kannan, R. and Vempala, S.: "Fast Monte-Carlo algorithms for finding low-rank approximations", *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, 1998.

[12] Golub, G. H. and Van Loan, C. F.: "*Matrix Computations*", The Johns Hopkins University Press, Third Edition, 1996.

[13] Letsche, T. A. and Berry, M. W.: "Large-scale information retrieval with latent semantic indexing", *Information Sciences – Applications*, 100, pp. 105-137, 1997.

[14] Jiang, F., Kannan, R., Littman, M. L. and Vempala, S.: "Efficient singular value decomposition via improved document sampling", Technical Report CS-99-5, Department of Computer Science, Duke University, 1999.

[15] Papadimitriou, C. H., Raghavan, P., Tamaki, H. and Vempala, S.: "Latent semantic indexing: A probabilistic analysis", *Proceedings of the 17th ACM-SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pp. 159-168, 1998.

[16] Press, W. H. at al.: "*Numerical Recipes in C : The Art of Scientific Computing*", Cambridge University Press, 1988.

[17] Zha, H. and Zhang, Z.: "On matrices with low-rank-plus-shift structures: Partial SVD and latent semantic indexing", Technical Report CSE-98-012, Department of Computer Science and Engineering, Pennsylvania State University, 1998.