特徴素の選択と集約による DSA のためのデータ要約

相澤 彰子
学術情報センター

情報検索において最も一般的なデータ表現である文書対用語頻度行列は，統計的には分割表と呼ばれるデータ表現の一種であるとみなせる．分割表において行および列間の関連度を評価するための基本的な統計分析法として双対尺度法（DSA）がある．本論文では，用語に対して分類階層があらかじめ与えられているものと仮定した上で，DSA を適用するための用語数削減の手法について検討する．具体的には，用語選択と用語集約と呼ぶ 2 つのデータ要約手法を考え，両者を比較するのための評価式を導出する．そして，分類階層上で定義された各用語集合に対して選択と集約のいずれかを評価式から数値的に判断して適用するデータ要約手順を提案し，その有効性を実際の文書コーパスと専門用語辞書を用いて示す．

A Method for Dimension Reduction for Dual Scaling Analysis by Selecting and Aggregating Features

Akiko AIZAWA
National Center for Science Information Systems

Document by term frequency matrices are the most common data representations in information retrieval which can be viewed as special type of contingency tables. For this type of data, *dual scaling analysis* (DSA) is known to be a fundamental statistical method to explore underlying association structure among rows and among columns. In this paper, we first assume a classification hierarchy is given for terms, and then calculate the comparative data losses for two reduction schemes, term selection and term aggregation, both are the methods for reducing the dimension of the original matrix for DSA. We also propose a new reduction procedure where the derived equations are used to decide which scheme should be employed for each term group on a given hierarchy. The effectiveness of the proposed method is demonstrated through experiments using actual text corpus and standard terminological dictionaries.

1 Introduction

Document by term frequency matrices or *lexical tables* are the most common data representations in information retrieval which can be viewed as special type of contingency tables whose cell (i, j) represents the number of occurrences of term j in document i. For this type of data representations, *dual scaling analysis* (DSA) [1] or *correspondence analysis* [2] is known to be a fundamental statistical method to explore the underlying association structure among rows and among columns. Though the usefulness of DSA was demonstrated in various information retrieval applications including automatic indexing [3] or concept space visualization [4], the computational cost prohibits the method to be applied to matrices with tens of thousands of terms. The problem thus becomes how to reduce the dimension of the original data at the pre-processing stage, to make DSA feasible.

In many information retrieval applications, it is a common practice to select significant terms simply based on the total frequency they appear in a whole document set, sometimes discarding top ranked ones since they most likely appear in all the documents and thus statistically meaningless. There also exist some studies in text-learning field concerning the selection of representative terms. As is pointed out in the past studies [5][6], selection procedures used in text-learning are simple compared with the ones developed for feature subset selection in machine learning or pattern recognition; in most cases, each terms are first evaluated independently using some statistical measures such as information gain or cross entropy, and then, a specified number of subset with highest rankings are selected. Such simple scheme enables to manipulate large number of terms collected from target text corpus and to reduce the matrix size for more computationally intensive analysis such as DSA.

The main purpose of feature subset selection in text-learning is to reduce the number of terms while maintaining associations among documents as close as possible to the original ones, rather than to improve the performance of a specific document classification task, as is the case in machine learning. In addition, feature subset selection in text-learning has one interesting aspect; a classification hierarchy is given for features, i.e. thesaurus constructed by human is often available. The existence of such classification hierarchy implies a possibility of reduction not only by eliminating non-significant terms but also by substituting a group of terms with a corresponding upper class term. In this paper, we refer to the former type of reduction scheme by *term* or *feature selection* and the later type by *term* or *feature aggregation*.

In order to illustrate the effect of DSA and term aggregation, let us consider document by term matrix given as follows:

$$X = \begin{array}{c} \\ d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \\ d_9 \\ d_{10} \end{array} \begin{array}{c} \begin{array}{cccccccccc} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 & t_9 & t_{10} \end{array} \\ \left[\begin{array}{cccccccccc} 1 & 3 & 4 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 5 & 20 & 54 & 2 & 0 & 0 & 3 & 5 & 0 & 0 \\ 0 & 45 & 3 & 0 & 55 & 1 & 1 & 1 & 1 & 1 \\ 3 & 9 & 5 & 2 & 2 & 0 & 0 & 0 & 0 & 1 \\ 55 & 3 & 36 & 48 & 4 & 0 & 2 & 0 & 2 & 1 \\ 1 & 3 & 1 & 1 & 1 & 56 & 34 & 33 & 21 & 3 \\ 0 & 0 & 8 & 0 & 6 & 30 & 4 & 5 & 33 & 0 \\ 1 & 0 & 3 & 0 & 0 & 0 & 12 & 18 & 19 & 18 \\ 5 & 2 & 2 & 0 & 0 & 20 & 5 & 29 & 0 & 30 \\ 0 & 2 & 0 & 2 & 1 & 10 & 5 & 9 & 9 & 0 \end{array}\right] \end{array}$$

Assume the first five terms (say {*cat*, *dog*, *rabbit*, *tiger*, *bear*}) are categorized into one group (*animals*) and the rest (say {*car*, *train*, *bus*, *ship*, *truck*}) to another (*vehicles*). Aggregating these two groups of terms reduces the dimension of the matrix from 10 to 2, generating:

$$X_1 = \left[\begin{array}{cccccccccc} 9 & 81 & 103 & 21 & 146 & 7 & 14 & 4 & 9 & 5 \\ 1 & 8 & 5 & 1 & 5 & 147 & 72 & 67 & 114 & 33 \end{array}\right]^T,$$

where each cell represents the total frequency of terms in the same group. On the other hand, selecting the top two most frequent terms across documents yields:

$$X_2 = \left[\begin{array}{cccccccccc} 0 & 0 & 1 & 0 & 0 & 56 & 30 & 0 & 20 & 10 \\ 1 & 5 & 1 & 0 & 0 & 33 & 5 & 18 & 59 & 9 \end{array}\right]^T.$$

Also, selecting the most frequent terms from each group produces:

$$X_3 = \left[\begin{array}{cccccccccc} 4 & 54 & 3 & 5 & 36 & 1 & 8 & 3 & 2 & 0 \\ 1 & 5 & 1 & 0 & 0 & 33 & 5 & 18 & 59 & 9 \end{array}\right]^T.$$

Table 1 shows the results of UPGMA clustering [7] where 10 documents are categorized into two groups using the distance calculated either from X, X_1, X_2 or X_3 either before or after dual scaling. It can be seen that (i)without DSA, neither of the methods able to identify expected clustering result $\{d_1,d_2,d_3,d_4,d_5\}$ $\{d_6,d_7,d_8,d_9,d_{10}\}$, and that (ii)with DSA, X and X_1 produce the same expected result while X_2 and X_3 still do not. Based on this, we can conclude that DSA with term aggregation works more successfully than term selection in this case.

Though term aggregation seems to be a natural choice to reduce the dimension of terms, the mathematical meanings of it has not been sufficiently examined in the past. This paper aims at providing mathematically justified criteria to decide which reduction schemes, term selection or term aggregation, should be employed for a given term group. For this purpose, we calculate the comparative data losses of the two reduction schemes. We also propose a new reduction procedure that combines these two.

Table 1: Result of UPGMA clustering.

(a) before dual scaling

X :	$\{d_1,d_2,d_3,d_4,d_6,d_7,d_8,d_9,d_{10}\}$	$\{d_5\}$
X_1 :	$\{d_1,d_2,d_3,d_4,d_5,d_7,d_8,d_{10}\}$	$\{d_6,d_9\}$
X_2 :	$\{d_1,d_2,d_3,d_4,d_5,d_7,d_8,d_{10}\}$	$\{d_6,d_9\}$
X_3 :	$\{d_1,d_3,d_4,d_6,d_7,d_8,d_9,d_{10}\}$	$\{d_2,d_5\}$

(b) after dual scaling

X :	$\{d_1,d_2,d_3,d_4,d_5\}$	$\{d_6,d_7,d_8,d_9,d_{10}\}$
X_1 :	$\{d_1,d_2,d_3,d_4,d_5\}$	$\{d_6,d_7,d_8,d_9,d_{10}\}$
X_2 :	$\{d_1,d_2,d_8\}$	$\{d_3,d_4,d_5,d_6,d_7,d_8,d_{10}\}$
X_3 :	$\{d_1,d_2,d_3,d_4,d_5,d_7\}$	$\{d_6,d_8,d_9,d_{10}\}$

The rest of the paper is organized as follows: Section 2 gives a mathematical description of the problem. Section 3 first provides equations for losses for selecting and aggregating features and then a procedure to minimize the total loss value. Section 4 shows the result of experimental study using full-text survey articles and thesauri of standard technical terms. Section 5 includes some discussions and future issues.

2 Problem Description

2.1 Dual Scaling Analysis

Let X be a $m \times n$ matrix with each cell x_{ij} representing the frequency of term j $(1 \leq j \leq n)$ in document i $(1 \leq i \leq m)$. DSA can be viewed as a variation of principle axes methods which determines the optimal scoring for rows and columns so that the correlation between these two is maximized. Mathematically, DSA adopts a general matrix conversion procedure called *singular value decomposition*, after transforming X to Y given by:

$$y_{ij} = \frac{x_{ij}}{\sqrt{x_{i\cdot}}\sqrt{x_{\cdot j}}} - \frac{\sqrt{x_{i\cdot}}\sqrt{x_{\cdot j}}}{x_t}, \qquad (1)$$

with $x_{i\cdot}$ and $x_{\cdot j}$ being the sums of row i and column j respectively, and x_t the sum of all x_{ij}. The computation complexity of DSA is essentially the one needed for matrix inversion.

The above transformation, together with the nature of singular value decomposition, guarantees that adding or eliminating columns from X does not affect DSA results for the remaining columns so far as the sum of rows $(x_{i\cdot})$ and the total sum (x_t) are maintained. This property of DSA, which we utilize later in our theoretical development, is called *the principle of equivalent partitioning.*

2.2 Classification Hierarchy of Terms

The representation of a *term classification hierarchy* in this paper is simply a tree whose leaves correspond to *basic* terms, terms actually observed in documents, and nodes to *term groups*, abstract terms defined by a group of basic terms. A basic term itself can be viewed as a special term group with a single group member. In the following, we use the lower case t for basic terms and the upper case T for term groups. For example, in Figure 1, T_1 consists of terms $\{t_3, t_4, t_5, t_6, t_7\}$, T_5 consists of a single term $\{t_5\}$, and so on.

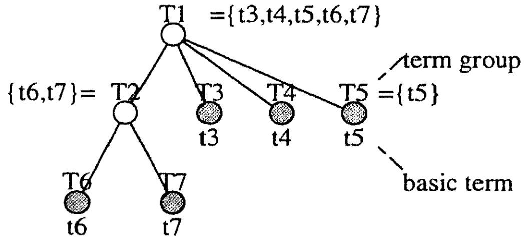


Figure 1: Classification Hierarchy of Terms.

It should be noted that a classification hierarchy represents only *a priori* knowledge about possible resemblance of terms and whether terms categorized into the same group actually show similar patterns remains to be tested on the target data.

2.3 Term Selection and Aggregation

Let $\vec{x}_{(j)}$ represent m dimensional vector corresponding to term t_j, i.e. $\vec{x}_{(j)} = (x_{1j}, \cdots, x_{mj})^T$, the j'th column of X. Given a term group T of size k, *term selection* refers to the case where k_0 $(\leq k)$ of the k terms is selected as representatives of T while the other $(k - k_0)$ terms are discarded. On the other hand, *term aggregation* refers to the case where the k term vectors in T are replaced with a single newly generated vector $\vec{x}$ such that

$$\vec{x} = (\sum_{j=1}^{k} x_{1j}, \cdots, \sum_{j=1}^{k} x_{mj})^T. \qquad (2)$$

Thus in term aggregation, only the total frequency of k terms is maintained after reduction while all other information is discarded. Here, we assume term aggregation occurs only at once and do not consider the situation when a subset of the group are aggregated.

3 Theoretical Development

3.1 Loss of Term Selection

Assume terms $\{t_{k+1}, \cdots, t_n\}$ are selected from the original matrix X, generating a reduced matrix $X' = (\vec{x}_{(k+1)}, \cdots, \vec{x}_{(n)})$. From the principle of equivalent partitioning, the reduction is, in terms of DSA, equivalent to substituting each of the eliminated terms $\{t_1, \cdots, t_k\}$ with an identical vector $\vec{x}' = \{x'_1, \cdots, x'_m\}$ given by:

$$x'_i = \frac{1}{\alpha} \sum_{j=k+1}^{m} x_{ij}, \; \alpha = \frac{k \sum_{i=1}^{m} \sum_{j=k+1}^{n} x_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{k} x_{ij}}. \qquad (3)$$

Note that in the above equation, the sum of each rows and the total frequency are maintained. Using $\vec{x}'$, we define the loss of selecting $\{t_{k+1}, \cdots, t_n\}$, i.e. the loss of eliminating $\{t_1, \cdots, t_k\}$ as follows:

$$\begin{aligned} L_{sel}(t_{k+1}, \cdots, t_n) &= L_{eli}(t_1, \cdots, t_k) \\ &= \sum_{i=1}^{m} \sum_{j=1}^{k} (x_{ij} - x'_i)^2 . (4) \end{aligned}$$

Though there may exist other possible equivalent transformations for X', Eq.(3) is used as a basis in this paper because the calculation of the optimal transformation is obviously infeasible and the equation has good rationale as it approximates the missing terms by the average of the remaining ones.

3.2 Loss of Term Aggregation

Loss of term aggregation on the other hand is calculated as follows: From the principle of equivalent partitioning, aggregating terms $\{t_1, \cdots, t_k\}$ to a single term given by Eq.(2) is, in terms of DSA, equivalent to substituting all the k terms with identical term vector $\vec{x}'' = \{x''_1, \cdots, x''_m\}$ such that:

$$x''_i = \frac{1}{k} \sum_{j=1}^{k} x_{ij}. \qquad (5)$$

Using $\vec{x}''$, we define the loss for term aggregation as follows:

$$L_{agg}(t_1, \cdots, t_k) = \sum_{i=1}^{m} \sum_{j=1}^{k} (x_{ij} - x''_{ij})^2. \qquad (6)$$

When the term group consists of only a single term ($k = 1$), the loss value equals zero.

Again, there may exist other better equivalent transformations in terms of the loss value, but we use Eq.(5) as our basis because of the simplicity of the calculation and semantic clearness; when aggregated, all the terms in the group are averaged. The effectiveness of such approximation is examined through experiments in the next section.

Figure 2 depicts the difference of term selection and aggregation where terms $\{t_1, \cdots, t_k\}$ are reduced into a single term.

3.3 Total Loss Calculation

Let $\mathcal{T}$ be a set of term groups obtained by applying selection and aggregation. Every basic term in X either (i)belongs to one and exactly one of the selected term groups (including the term itself), or (ii)otherwise. Then, the overall loss can be calculated as follows:

$$\begin{aligned} L_{total}(\mathcal{T}) &= L_{sel}(\{t_j \mid t_j \in T, T \in \mathcal{T}\}) \\ &\quad + \sum_{T \in \mathcal{T}} L_{agg}(\{t_j \mid t_j \in T\}).(7) \end{aligned}$$

3.4 Procedure for Reduction

Assume n_0 be a specified number of terms for reduction. The procedure we propose here is a

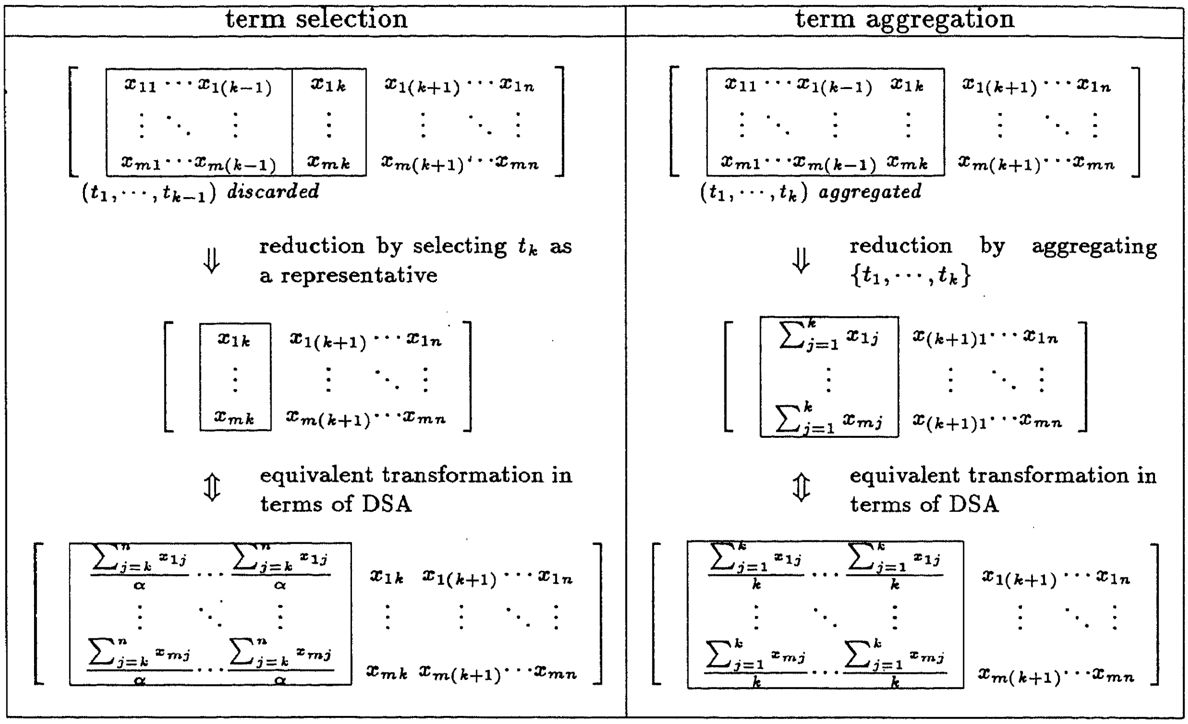

term selection
term aggregation
$(t_1, \cdots, t_{k-1})$ *discarded*
$(t_1, \cdots, t_k)$ *aggregated*
reduction by selecting t_k as a representative
reduction by aggregating $\{t_1, \cdots, t_k\}$
equivalent transformation in terms of DSA
equivalent transformation in terms of DSA

Figure 2: Reducing $\{t_1, \cdots, t_k\}$ into a single term by selection and aggregation

sort of greedy algorithm which minimizes the total loss given by Eq.(7). While its execution, it maintains a list of sorted term groups and assumes the top n_0 term groups survive for selection. In our current implementation, the loss is optimized by the following steps :

step(1): Initially, sort all the basic terms in X according to their elimination loss given by Eq.(4). Select the top n_0 basic terms and calculate the total loss of the selection.

step(2): Chose one of the upper term groups which does not appears on the list but has its members already listed. Calculate the total loss assuming its members are aggregated.

step(3): If the loss value improves, eliminate all the group members and add the upper term to the sorted list.

step(4): Repeat step (2) and (3) until all the upper groups are tested.

Figure 3 illustrates the proposed reduction procedure.

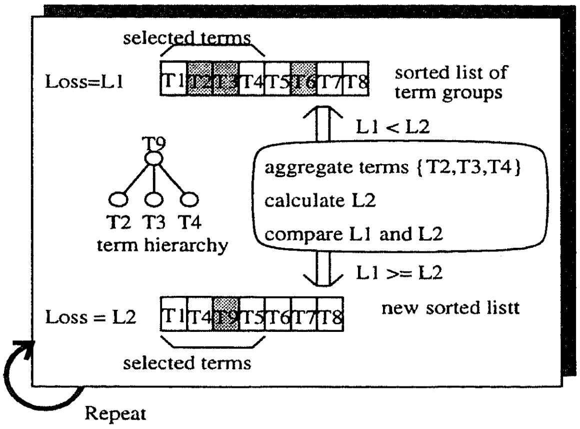


Figure 3: Classification Hierarchy of Terms.

Note that the procedure does not require much computation cost; the loss calculation at Step(2) is in proportional to the size of the matrix, and the iteration of Step(2)-(3) is no more than the number of non-basic terms, which is relatively small as will be seen in Table 2.

4 Experimental Results

4.1 Target Data

The information sources we use in our experiment are the survey articles in *Journal of Japanese Society for Artificial Intelligence* published during 1986-1995. The total 304 articles exist as full-text, digitized by OCR under NACSIS Corpus Project [8] [9].

We compare three different term hierarchies extracted from well-known terminological dictionaries: (i)the one obtained from a technical term dictionary provided by Japanese Society for Artificial Intelligence [11] (*JSAI*), (ii)the one obtained from Technical Terms Dictionary of EDR Electronic Dictionary Version 1.5 [10] (*EDR*), and (iii)(i) expanded with synonymous term corpus automatically generated [12] (*JSAI**). Since JSAI is constructed by the society itself, we can expect it is the most suitable to characterize the target document set. On the other hand, EDR contains a large number of terms relating information processing in general and thus may not be so much focused on the subject domain. In JSAI*, each term in JSAI is expanded with a group of corresponding synonyms including various notation variations.

Technical terms defined in each dictionary constitute a set of basic terms. The occurrence of each basic term is counted for each article to make the initial frequency matrix $F = \{f_{ij}\}$. While counting, only simple normalization rules such as ignoring lower and upper case difference are applied for JSAI and EDR while in JSAI*, the occurrence counter is increased every time when one of the synonymous terms appears. Thus, the process can be considered as *enforced* aggregation where term aggregation is always applied at the bottom level. The difference between JSAI and JSAI* is, for example, in JSAI, the Japanese word "metafa(*metaphor* in English)" and English word "metaphor" (Figure 4) are considered different and thus assigned different frequencies. On the other hand, these two as well as other terms as "metaphorical expression", "in-yu(*metaphor*)", "hiyu(*metaphor*)" are considered to be the same in JSAI* and thus assigned the same frequency.

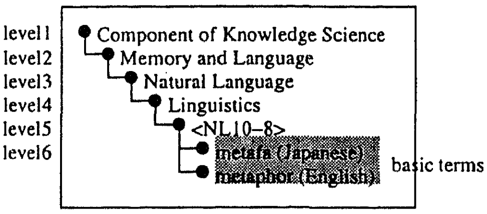


Figure 4: Example of terms defined by JSAI.

Table 2 shows the comparison of JSAI, EDR and JSAI*. It can be seen that EDR contains wide range of terms with less occurrences. Also, the average frequency per term for JSAI* is greater than the one for JSAI because of the term expansion effect. For reference, values without such expansion are also shown in the table by figures with brackets.

Table 2: Comparison of JSAI, EDR and JSAI*.

	JSAI	EDR	JSAI*
total number of basic terms	2,098	195,179	2,098 (10,265)
total number of term groups †	3,329	197,307	3,329 (13,594)
average depth of term hierarchy	6.0	7.7	6.0 (7.0)
number of terms actually observed	483	12,026	1,104 (3,196)
average frequency per observed term	0.56	0.25	1.29 (0.45)

† including basic terms

4.2 Method for Evaluation

Once the initial frequency matrix is obtained, each row of the matrix is normalized so that the total frequency for each document equals 1.0, i.e. $x_{ij} = f_{ij}/\sum_{j=1}^{n} f_{ij}$. Term reduction is then performed to reduce the size of the matrix to a specified number.

We compare the proposed reduction scheme (*LRED, reduction-by-loss*) with a simple selection without aggregation where basic terms are evaluated and selected using elimination loss given by Eq.(4) (*LSEL, selection-by-loss*), and also with a naive but commonly practiced method that selects the top frequent terms (*FSEL, selection-by-frequency*). We have also tested selection using cross entropy, but simple frequency worked better in our case. This may be because the authorized terminological sets do not include meaningless words such as stop words which cross entropy method can be the most effectively applied to.

After the reduction, DSA is applied to each reduced matrix to calculate similarities between every document pair. As reference, we also directly apply DSA to the original normalized (but not reduced) matrix and calculate similarities. The performance of LRED, LSEL and FSEL is evaluated by the correlation between these similarity values. Pearson's product moment correlation coefficient is used to see the correlation in terms of values, and Spearman's rank correlation coefficient in terms of ranks. For both coefficients, the value 1.0 shows the most fittest.

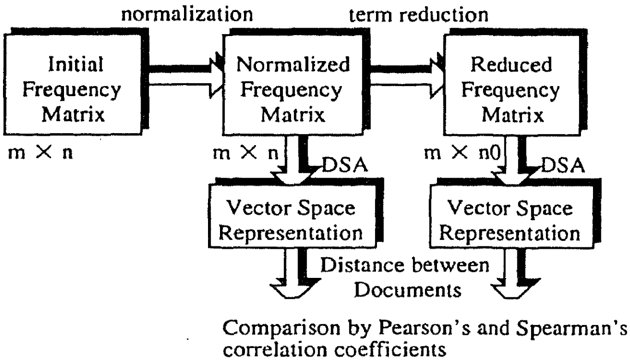
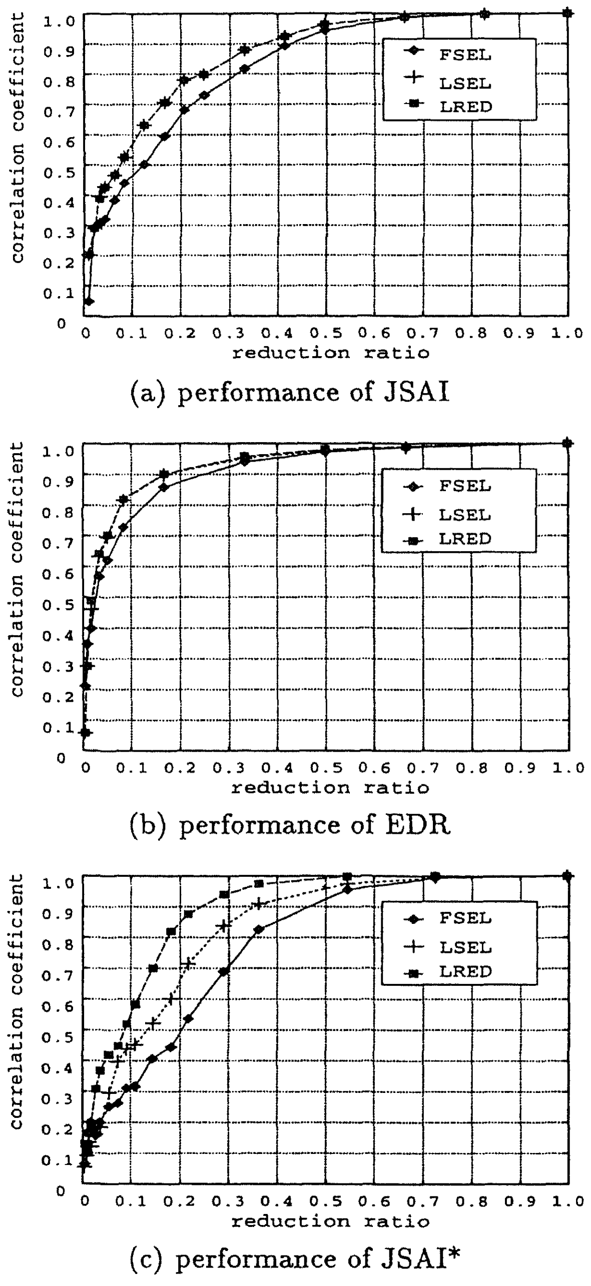

normalization
term reduction
Initial Frequency Matrix
Normalized Frequency Matrix
Reduced Frequency Matrix
m X n
m X n
DSA
m X n0
DSA
Vector Space Representation
Vector Space Representation
Distance between Documents
Comparison by Pearson's and Spearman's correlation coefficients

Figure 5: Evaluation method.

4.3 Results

Figure 6 compares the performance of FSEL, LSEL and LRED for each of JSAI, EDR, and JSAI*. The horizontal axis represents the ratio of selected terms to the total terms actually observed, and the vertical axis to the corresponding values of rank correlations. Results of distance correlations are omitted since these two showed exactly the same tendency.



(a) performance of JSAI

(b) performance of EDR

(c) performance of JSAI*

Figure 6: Comparison of FSEL, LSEL and LRED for JSAI, EDR and JSAI*.

The experiment shows that LRED and LSEL outperform FSEL in all cases, justifying the use of the loss equation as a selection criteria. Also, LRED is better than LSEL, demonstrating that selection by loss and term aggregation both contribute to the performance improvement. Next, comparison across different term hierarchies suggests the aggregation effect depends on the kind of thesaurus adopted. Only small improvement is observed for JSAI and EDR since they do not contain much redundancy as JSAI*. Especially for JSAI, only several terms are aggregated which means most of the improvement is obtained by using the proposed loss equation for term selection. The result for JSAI* shows that 'enforced' aggregation to some extent works well. This can be explained as follows: the reduction scheme we have developed totally depends on statistical characteristics of terms, and thus can hardly deal with synonymous problem where the same concept is referred to with different terms, due to the language-difference, the use of acronyms and so on.

5 Discussions

In the above experiments, we use only limited size of data in order to make the original matrix feasible for DSA. However, term aggregation itself is applicable for larger sizes. Variations of the proposed method with less computation overhead are examined using real-scale HTTP log data [13] [14].

All of (i)the conventional simple selection methods, (ii)selection with aggregation examined in this paper, and (iii)DSA can be utilized for reduciong dimension of terms. The difference is that the dependencies considered among terms are either (i)on an individual basis, (ii)within pre-determined term groups, or (iii)among all terms. Correspondingly, DSA requires much more computation time and results in better approximation. At the same time, DSA has one problem inherent to principle axes methods; that is, the interpretation of the generated terms (principle axes) is often difficult. Since the proposed method simply substitutes a group of terms with a well-defined upper class concept, the method is sometimes preferable than DSA especially when is used as pre-processing filter for other IR applications.

Acknowledgements

The authors would like to thank Kenro Aihara, Atsushi Takasu and other colleagues at National Center for Science Information Systems for helpful discussions.

References

[1] Nishisato,S.: "Analysis of Categorical Data: Dual Scaling and It's Applications, " Asakura-Shoten (1982).

[2] Lebart,L., Salem,A. and Berry,L.: "Exploring Textual Data," Kluwer Academic Publishers (1998).

[3] Deerwester,S., Dumais,S.T., Furnas,G.W., Landauer,T.K. and Harshman,R. : "Indexing by Latent Semantic Analysis," Journal of the American Society of Information Science, Vol,41, No.6, pp.391-407 (1990).

[4] Sugimoto,M., Hori,K. and Ohsuga,S.: "A System for Visualizing Viewpoints and Its Application to Intelligent Activity Support," *IEEE Trans. System, Man and Cybernetics*, Vol.28C, No.1, pp.124-136 (1998).

[5] Mladenic,D.: "Feature Subset Selection in Text-Learning," 10th European Conference on Machine Learning (1998).

[6] Yang,Y. and Pedersen,J.O.: "A Comparative Study on Feature Selectino in Text Cagegorization," Proc. of the 14th International Conference on Machine Learning, p.412-420 (1997).

[7] Kaufman,L. and Rousseeuw,P.J.: "Finding Groups in DATA," Wiley-Interscience (1990).

[8] Aihara,K. and Takasu,A.: "Domain Visualization Based on Authorized Documents," proceedings of the fourth International Conference on Information Systems, Analysis and Synthesis, p.391-398 (1998).

[9] Kageura,K., Koyama,T., Yoshioka,M., Takasu,A., Nozue,T. and Tsuji,K.: "NACSIS Corpus Project for IR and Terminological Research," NLPRS'97, p.493-496 (1997).

[10] EDR: "EDR Electronic Dictionary Version 1.5 Technical Guide," TR2-007, Japan Electronic Dictionary Research Institute (1996).

[11] Japan Society for Artificial Intelligence (ed.): "Jinko Chinou Handobukku (Handbook of Artificial Intelligence)," Ohm-sha (1990).

[12] Aizawa,A. and Kageura,K.: "An Approach to the Automatic Generation of Multilingual Keyword Clusters," COMPTERM'98 (1998).

[13] Aizawa,A.: "Analysis of Internet Domains by Extracting Information from HTTP Logfiles," Journal of IEICE, Vol.J81-DI, 1988-11 (1998).

[14] Aizawa,A.: " Reducing the Dimensions of Attributes by Selection and Aggregation (Abstract)," The First International Conference on Discovery Science (1998) (to be presented).