Neural Processing Letters **19:** 157–173, 2004.

LSISOM – A Latent Semantic Indexing Approach to Self-Organizing Maps of Document Collections

NIKOLAOS AMPAZIS[1] and STAVROS J. PERANTONIS[2]
[1]*Department of Financial and Management Engineering, University of the Aegean, 82100, Chios Greece. e-mail:n.ampazis@fme.aegean.gr*
[2]*National Center for Scientic Research 'Demokritos', Institute of Informatics and Telecommunications, Ag. Paraskevi Attikis, P.O. Box 60228, Athens, 15310, Greece. e-mail: sper@iit.demokritos.gr*

Abstract. The Self Organizing Map (SOM) algorithm has been utilized, with much success, in a variety of applications for the automatic organization of full-text document collections. A great advantage of the SOM method is that document collections can be ordered in such a way so that documents with similar content are positioned at nearby locations of the 2-dimensional SOM lattice. The resulting ordered map thus presents a general view of the document collection which helps the exploration of information contained in the whole document space. The most notable example of such an application is the WEBSOM method where the document collection is ordered onto a map by utilizing word category histograms for representing the documents data vectors. In this paper, we introduce the LSISOM method which resembles WEBSOM in the sense that the document maps are generated from word category histograms rather than simple histograms of the words. However, a major difference between the two methods is that in WEBSOM the word category histograms are formed using statistical information of short word contexts whereas in LSISOM these histograms are obtained from the SOM clustering of the Latent Semantic Indexing representation of document terms.

Key words. data representation, document clustering, information retrieval, latent semantic indexing, self-organizing maps, unsupervised learning

1. Introduction

The utilization of Self Organizing Maps (SOMs) for the automatic organization of full-text document collections has been shown to provide an invaluable aid to traditional Information Retrieval (IR) systems [5, 14, 18]. SOMs have the ability to arrange documents with similar content in neighboring regions which, by analogy, is comparable to the situation encountered in conventional libraries, where books are organized in thematic topics. Such an arrangement, combined with traditional IR search tools and facilities, can help users not only to search for a specific piece of information and to retrieve documents within one topical cluster, but also to get an overview of the whole document collection and to explore the extend to which the topic of their interest is covered.

Since the early 90's, there have existed attempts [20, 25, 27] to apply the SOM in the textual domain, based on the encoding of documents according to the Vector

Space Model (VSM) [24]. The VSM encodes documents in a textual collection as vectors in a multidimensional feature space. In this space each dimension corresponds to one word and the value of each vector component is a function of the frequency of occurence of that particular word (word histogram vectors). It is obvious that in such a representation, the dimensionality of the resulting document vectors is very high since it depends on the size of the vocabulary used in the entire document collection. In order to avoid such high dimensionalities, the vocabularies are usually limited manually. However, in order to classify masses of natural texts, it is usually unavoidable to refer to a rather large vocabulary size. There exist at least four possibilities to reduce the dimensionalities of the histogram vectors, without significantly lowering the corresponding quality of clustering:

- Projection of the data onto a lower-dimensional orthogonal subspace where most of the variance is concentrated in the new subspace's axes. For textual data domains this method is known as Latent Semantic Indexing (LSI) [7] and it is based on the Singular Value Decomposition (SVD) of the term-document matrix of the textual collection. In addition to dimensionality reduction, LSI exhibits improved retrieval performance since theoretical [8, 21] and experimental results [3] have shown that it enhances the semantic aspects of the data. A potential problem of LSI arises from its computational cost since the evaluation of the SVD for high dimensional data sets can be quite high.
- Projection of the original data onto a lower-dimensional subspace through the multiplication of the term-document matrix with a random matrix (Random Projection – RP method) [13]. Despite its computational simplicity, it has been shown both theoretically [13] and experimentally [4] that RP does not distort distances between points in the original data space especially in the case where the matrix to be projected is sparse.
- Reduction of the dimensionality of the histogram vectors by a composite Random Projection/Latent Semantic Indexing (RP/LSI) method which consists of an initial application of RP, that suitably reduces the original space dimension, which is then followed by LSI [21]. The main advantage of this method is that it benefits both from the computational simplicity of RP and the semantic enhancement of data of LSI.
- Clustering of words into semantic categories, as is done in the WEBSOM method [9, 10, 14, 17, 18].

The evaluation of the effects of the first three dimensionality reduction techniques on the ability of the SOM to semantically cluster textual data has been studied in an earlier publication of the authors of the present article [1]. In this article we describe a methodology, similar to the WEBSOM, for the dimensionality reduction of the original word histograms feature space, through the clustering of words into semantic categories. The proposed LSISOM method utilizes a SOM to cluster documents which are represented by word category histograms that are formed from a separate SOM clustering of the LSI representations of document terms.

2. LSISOM Method

The problem addressed by the LSISOM method is to automatically organize full-text document collections using a SOM, in order to enable the examination of the distribution of topics within the entire corpus. The high dimensionality of the VSM word histograms document representation, however, is a potential problem since it results in burdensome computations for the training of the SOM. It is therefore, beneficial to attempt a reduction in the dimensionality of the data vectors before the application of the SOM clustering algorithm which is based on the computation of distances in the feature space. An additional problem with the word histograms is that each word, irrespective of its meaning, contributes equally to the histogram. In other words, the VSM treats terms that happen to have similar meaning (synonymous expressions) in exactly the same way that it treats unrelated terms. A standard technique used in IR systems, in order to address this problem, is the utilization of thesauri in order to group terms that are conceptually related. There are two types of thesauri, manual and automatic. The major problem with manual thesauri is that they are expensive to build and hard to update in a timely manner [12]. Automatic thesauri are typically built based on co-occurrence information, and relevance judgements are often used to estimate the probability that a thesaurus term is similar to another term. Because relevance judgements are not always available, often these approaches are impractical for term classification or thesaurus construction. Second, even if available, relevance judgments are usually produced for a small set of terms, which does not cover the whole document collection [12].

In the WEBSOM method an alternative techique, employed for the semantic clustering of terms, is based on the statistics of the words contexts in order to provide information on their conceptual similarity. The clustering of terms results in a dimensionality reduction which is a fraction of the size of the original word histograms. This reduction is achieved through the utilization of the so-called 'self organizing semantic maps' [23] which are trained with the vectors of term statistics. However, a serious drawback of this approach is that the computation of the left and right context of each and every word that appears in any of the documents in the text corpus requires enormous computational resources. This shortcoming has recently led the developers of the WEBSOM to abandon the word contexts approach in favour of the RP method [16].

Similarly to WEBSOM, the proposed LSISOM method utilizes a self organizing semantic map to cluster individual terms into groups of similar concepts. In LSISOM, however, the map is trained with term vectors that are obtained from the semantically enhanced LSI representations of the document terms. Consequently, documents are represented by word category histograms rather than simple word histograms which results in significant dimensionality reduction of the original feature space. The details of the method are described in the following sections.

2.1. LATENT SEMANTIC INDEXING

LSI is a technique for substituting the original data vectors with shorter vectors in which semantic information is preserved but the effects of term usage variations are reduced. Because of the tremendous diversity in the words people use to describe the same object or concept (synonymy), similar concepts in different documents will often be described in a different way and the relevancy between them may be neglected. Conversely, since the same word often has more than one meaning (polysemy), irrelevant documents may become associated with each other. LSI achieves a reduction of these effects by constructing a linear mapping from the space spanned by the original VSM document vectors to a reduced dimensional subspace. This mapping is based on the SVD of the original *mxn* term-document matrix $\boldsymbol{A}$ i.e., the matrix of n, m-dimensional documents

$$\boldsymbol{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^T \tag{1}$$

where the orthogonal matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ contain the left and right singular vectors of $\boldsymbol{A}$ and the diagonal matrix Σ contains its singular values (Figure 1). LSI achieves the reduction in the dimensionality of the data by retaining only the k-largest ($k < r = \text{rank}(\boldsymbol{A})$) singular triplets of the decomposition of $\boldsymbol{A}$ which means that all data vectors $\boldsymbol{a}^i$ (columns of $\boldsymbol{A}$) are projected onto a k-dimensional subspace spanned by the left singular vectors corresponding to the k-largest singular values via the transformation

$$\hat{\boldsymbol{a}}^i = (\boldsymbol{a}^i)^T U_k \Sigma_k^{-1} \tag{2}$$

where $\boldsymbol{U}_k$ is of size *mxk* and contains these k singular vectors and Σ_k is of size *kxk* and contains the k largest singular values in its diagonal. In this sense the rows of $\boldsymbol{V}_k$ are considered as the LSI representations of the document vectors and, by an analogous argument, the rows of matrix $\boldsymbol{U}_k$ are considered as the LSI representations of the term vectors (Figure 1).

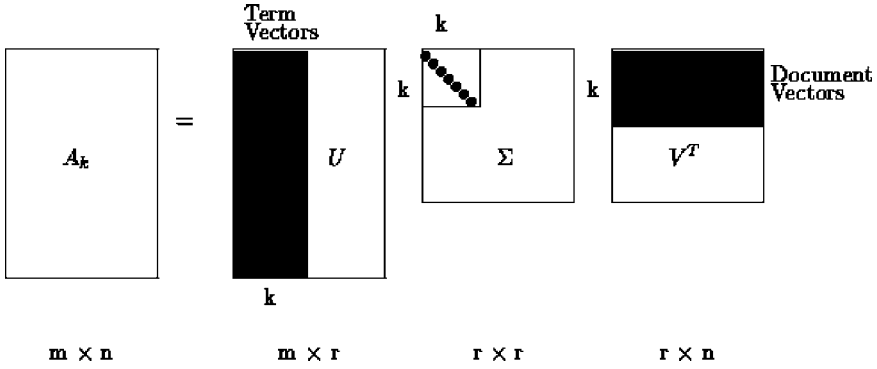

Figure 1. Singular value decomposition of the term-document matrix $\boldsymbol{A}$.

A key insight in LSI is that just as a document is represented by a vector of term frequencies (columns of $\boldsymbol{A}$), a term can be represented as a vector of document frequencies (rows of $\boldsymbol{A}$). These row vectors succinctly summarize everything that is revealed about the terms by the vectors which describe the document collection. Computing cosine similarities, for example, between these row vectors can reveal that some terms are used in a similar manner within the entire document collection. Large document collections provide more fine-grained term representations and the correspondence between semantic similarity and usage pattern similarity is, usually, sufficiently strong to automatically extract semantic information from these patterns.

The LSI representation of the term vectors (rows of $\boldsymbol{U}_k$) not only identifies similarities in the way terms are used in the collection but also suppresses the effect of term usage variations. This is achieved by assigning similar vectors to terms with similar usage, and dissimilar vectors to terms with significantly different usage. Maintaining the first k dimensions generally moves terms with similar meaning closer together and terms with dissimilar meanings remain far apart in the lower dimensional space [7]. Thus, the effectiveness of LSI relies on the ability of the SVD to extract salient features from the term frequencies across the entire set of documents in order to merge similar terms towards a single 'conceptual' representation. In a sense, the clustering of the LSI term representations with a SOM, that we propose in the LSISOM methodology, is hence similar to the way a human might choose to categorize two slightly different terms under the heading of a broader term when constructing a thesaurus.

2.2. SELF ORGANIZING MAPS

SOMs are unsupervised learning neural networks which were introduced by Kohonen [15] in the early '80s. This type of neural network is usually a two-dimensional lattice of neurons all of which have a reference model weight vector (Figure 2). As a result of the SOM training algorithm, these reference vectors (otherwise known as codebook vectors) are fitted to a set of input vectors by approximating the model of the data distribution in the high-dimensional document feature space. Therefore, the model vectors of neighboring units gradually learn to represent similar input data vectors.

SOMs are very well suited to organize and visualize complex data in a twodimensional display, and by the same effect, to create abstractions or clusters of that data. Therefore SOMs are frequently used in data exploration applications, but there exists a multitude of other applications as well [15].

The training of the SOM is achieved through a competitive learning process which consists of two steps that are applied iteratively. In the first step each input vector is compared to all the neurons' codebook vectors. The neuron s that has its codebook vector at the shortest geometric distance to an input vector, becomes the winner for that input vector. In the second step, each winning neuron and its surrounding neurons, i.e., neurons within a neighbourhood N_s gradually change the value of their codebook vectors in an attempt to match the input vector for which it has won. This cycle of competition and learning processes is repeated. At each cycle the size of the

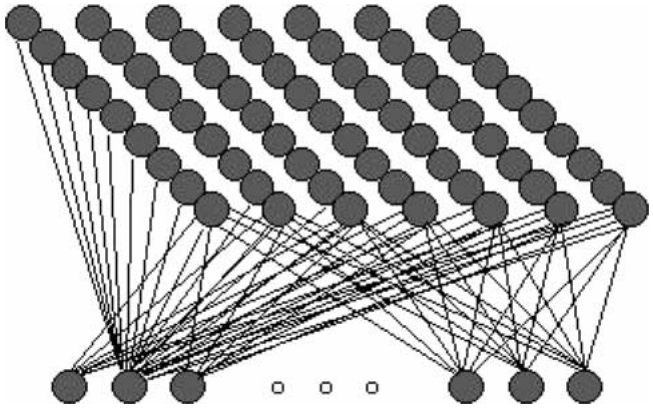

Figure 2. A two-dimensional self organizing map.

neighborhood of the winning neuron is decreased. The whole process terminates when each codebook vector has reached a satisfactory approximation of their corresponding input vector.

The steps of the SOM algorithm can be summarized as follows:

Step 1: Initialize
- weights to small random values
- neighbourhood size $N_s(0)$ to be large (but less than the number of neurons in one dimension of the array)
- parameter functions $a(t)$ and $\sigma^2(t)$ to be between 0 and 1

Step 2: Present an input pattern $\boldsymbol{x}$ through the input layer and calculate the Euclidian distance between the input vector and each weight vector:

$$d_j(t) = ||\boldsymbol{x}(t) - \boldsymbol{w}_j(t)|| = \sqrt{\sum_{i=1}^{n} (x_i(t) - \boldsymbol{w}_{ij}(t))^2}$$

Step 3: Select the neuron with minimum distance as the winner s.

Step 4: Update the weights connecting the input layer to the winning neuron and its neighbouring neurons (neurons k) according to the learning rule

$$\boldsymbol{w}_k(t+1) = \boldsymbol{w}_k(t) + c[\boldsymbol{x}(t) - \boldsymbol{w}_k(t)],$$

where $c = a(t)\exp(-||\boldsymbol{r}_i - \boldsymbol{r}_s||/\sigma^2(t))$ for all neurons j in $N_s(t)$, and $\boldsymbol{r}_i - \boldsymbol{r}_s$ is the physical distance (number of neurons) between neuron i and the winning neuron s.

Step 5: Continue from Step 2 for T epochs; then decrease neighbourhood size, $a(t)$ and $\sigma^2(t)$: Repeat until weights have stabilized.

In [15] it has been proved that the SOM algorithm always converges to a solution, i.e., that each of the winner weight vectors of the map converges to the mean of the data vectors for which it has been a winner, in a finite number of steps.

2.3. THE LSISOM ALGORITHM

The proposed algorithm, in essence, permits clustering of documents into the matically related groups by utilizing a significant dimensionality reduction of the original term (word) histograms feature space through the clustering of terms into semantic categories. To do so, the proposed method utilizes a two stage SOM clustering procedure: In the first stage a SOM ('self-organizing semantic map') is used to cluster the LSI representations of document terms (rows of matrix $\boldsymbol{U}_k$) into word categories as explained in Section 2.1. In the second stage, a different SOM is utilized in order to cluster the documents which are re-encoded by mapping their text, word by word, onto the first stage SOM.

The steps of the proposed algorithm can be summarized as follows:

Step 1: Utilize the Lanczos method (see Section 4) to obtain the SVD of the original sparse *mxn* term-document matrix $\boldsymbol{A}$, keeping the k largest singular components.

Step 2: Store the *mxk* matrix $\boldsymbol{U}_k$ whose rows are the LSI representations of the original term vectors.

Step 3: Use the rows of the matrix $\boldsymbol{U}_k$ as input data vectors to a SOM of fixed topology in order to directly cluster the terms.

Step 4: Train the SOM until convergence and re-encode the original documents by mapping their text, word by word, onto the SOM by locating the Best Mathing Unit (BMU) for each term on the map.

Step 5: Use the new representations of the documents as input data vectors to a new SOM of fixed topology in order to cluster the documents.

2.4. BENCHMARK DATA SET AND TEXT PREPROCESSING

For our experiments we used the 'Time Magazine' article collection which consists of 420 articles from the TIME Magazine from the 1960's. The complete collection can be obtained from the Internet URL address: 'http://www.ifs.tuwien.ac.at/~andi/somlib/data/time60/'.

At the same address there is also available, for downloading, a sample data set of the VSM representations of the collection that consists of 420, 5923-dimensional data vectors (i.e., 5923 distinct terms) which we used for our experiments. For this data set, the words of each document have been reduced to their rough stems by removing the most common suffixes, such as '-ed', '-ing' and plural (i.e. trailing '-s'). In addition, all words that appear in more than 90% of the documents have been removed, since these words do not contribute to content separation. This automatically eliminates words frequently referred to as 'stop words', such as articles, pronouns etc. ('the', 'is', 'are', 'he', 'she'), and therefore no manually constructed stop-word list was used. A further reduction in the vocabulary size has been achieved through the removal of all words that appeared in less than 3 documents, since such words provide only a very fine-grained content separation between the respective

documents. This has also helped in removing possible spelling errors. Finally, the calculation of the value of each vector component was based on the standard *tf x idf* [24] weighting scheme. A full account of the details of the text preprocessing methodology can be found at 'http://www.ifs.tuwien.ac.at/~andi/somlib/textrepresentation.html'.

2.5. WORD CATEGORY MAP

The word category map is the 'self-organizing semantic map' that describes the relations of terms based on their LSI representations, i.e, the rows of matrix U_k. For our experiments, we trained a rectangular SOM consisting of 15 by 21 nodes which is the standard word category map size used in WEBSOM. The value of k used for LSI was set to 100 and thus the training set consisted of the 5923, 100-dimensional LSI term vectors. The SOM is labeled after the training process by inputting each term vector once again to the trained word category map and recording their Best Matching Units (BMUs) on the map. Using this method a unit may become labeled by several terms, often synonymous or forming a closed attribute set [9]. Since the LSI representation of term vectors generally moves terms with similar meaning closer together and the SOM organizes similar input vectors in neighboring regions, interrelated words within the context of the document collection appear close to each other on the map. Thus, on the word category map similar words tend to occur in the same or nearby map nodes, forming 'word categories' in the nodes. The word category map is illustrated in Figures 3 and 4 which show a sample of the resulting categories (those formed at the 84 upper nodes of the map) due to space limitations. The map was computed using a standard Pentium III 866 MHz PC with 256MB of RAM using the Matlab SOM Toolbox [26].

2.6. DOCUMENT MAP

The 420 documents were encoded by mapping their text, word by word, onto the word category map. Hence, the original 5923-dimensional data vectors were substituted by 315-dimensional vectors (15×21) whose components were formed by the histograms of the BMUs of each word onto the word category map. Unlike the methodology adopted in WEBSOM the histograms were not blurred but instead each data vector was simply normalized to unit length. The document map was then formed by training a 10×15 SOM to cluster the various news articles (documents) by topic on the map. Figures 5 and 6 show the arrangement of the documents on the map. The clustering can be verified by reading the news articles located on identical or neighboring units. It is interesting to note that the document clustering is quite similar to the one reported in [22] where the original data set of 5923 dimensions was used to train a SOM (this issue is further discussed in Section 3). A full interpretation of the map trained with the original data set can be found in [22] and at the URL 'http://www.ifs.tuwien.ac.at/~andi/somlib/experimentstime60.html'.

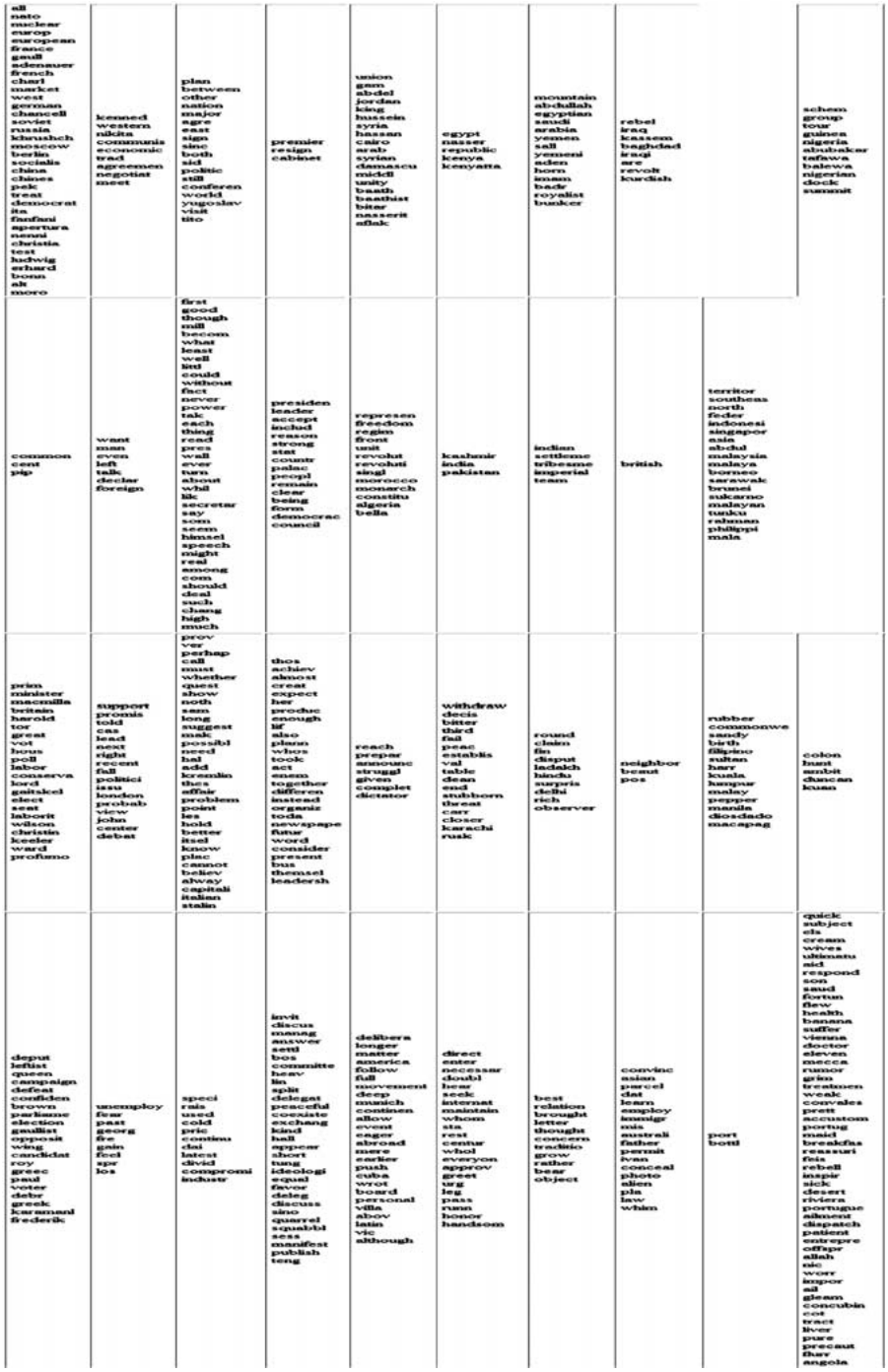

Figure 3. Word clusters formed on the 40 upper left nodes of the 15×21 word category map.

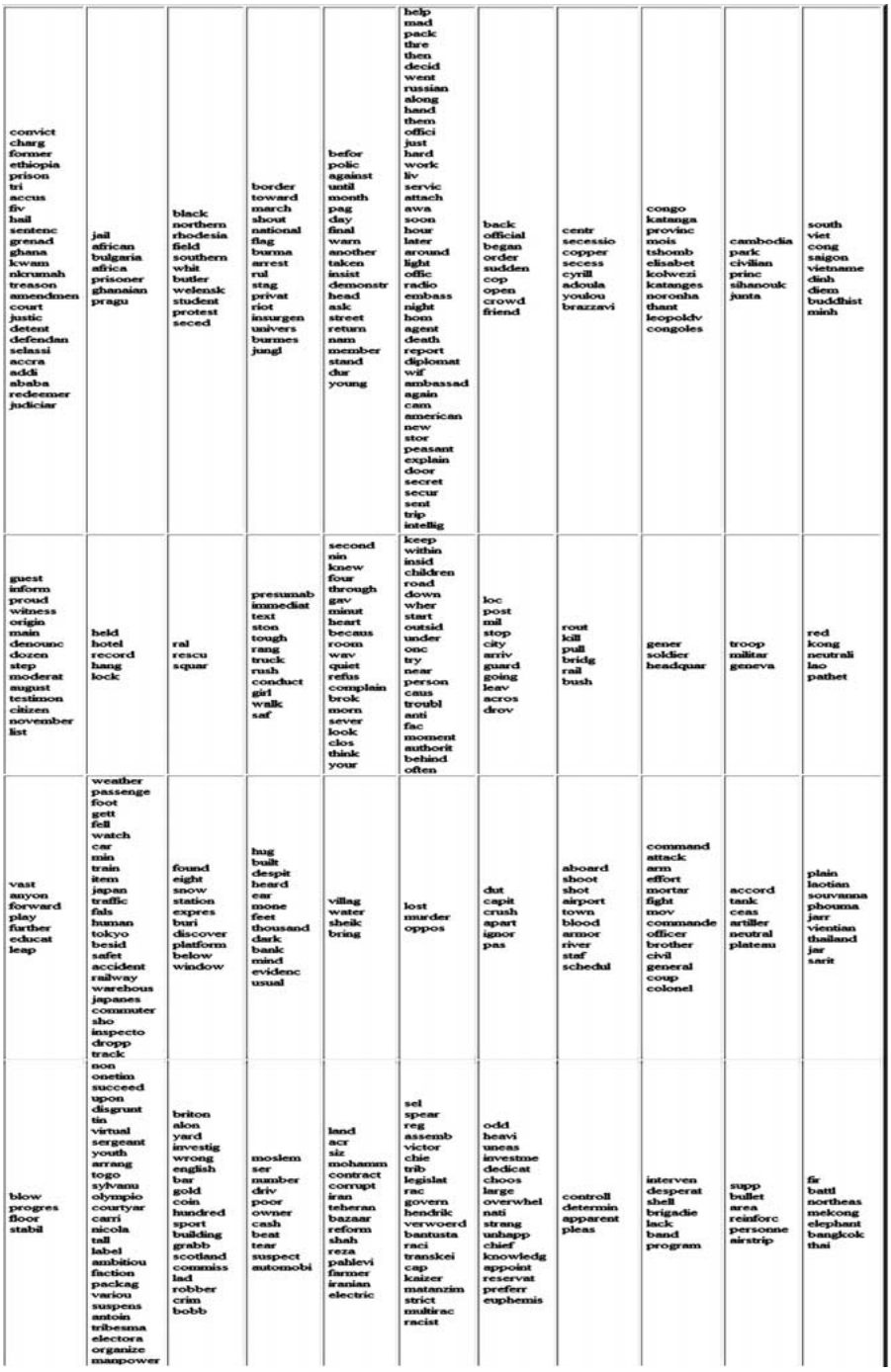

Figure 4. Word clusters formed on the 44 upper right nodes of the 15×21 word category map.

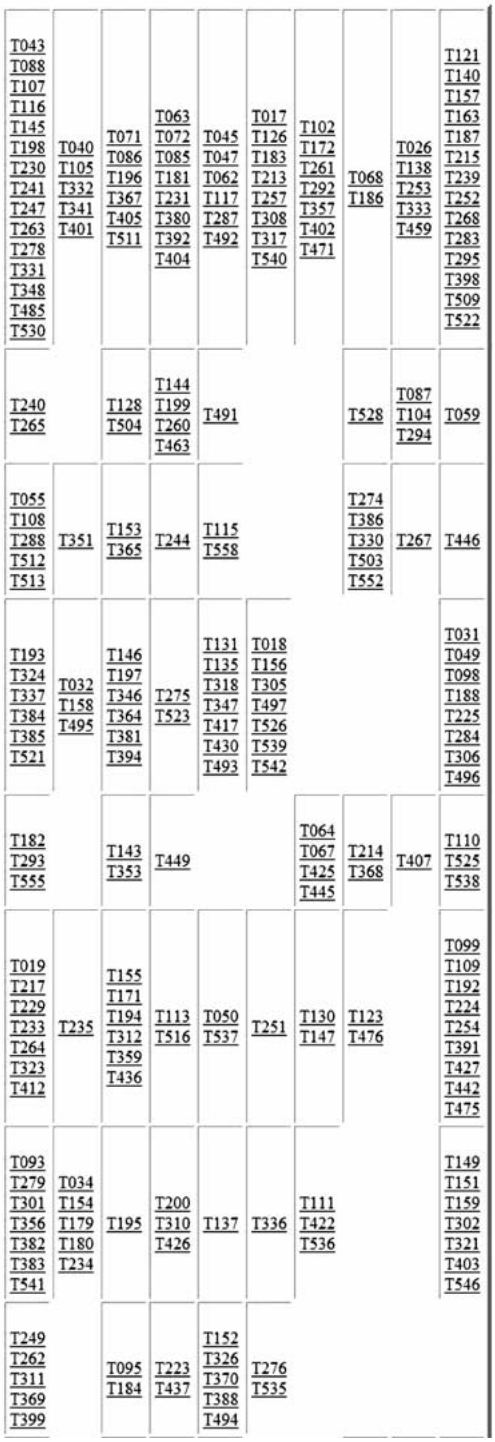

Figure 5. Document clusters on the upper half part of the 10×15 document map.

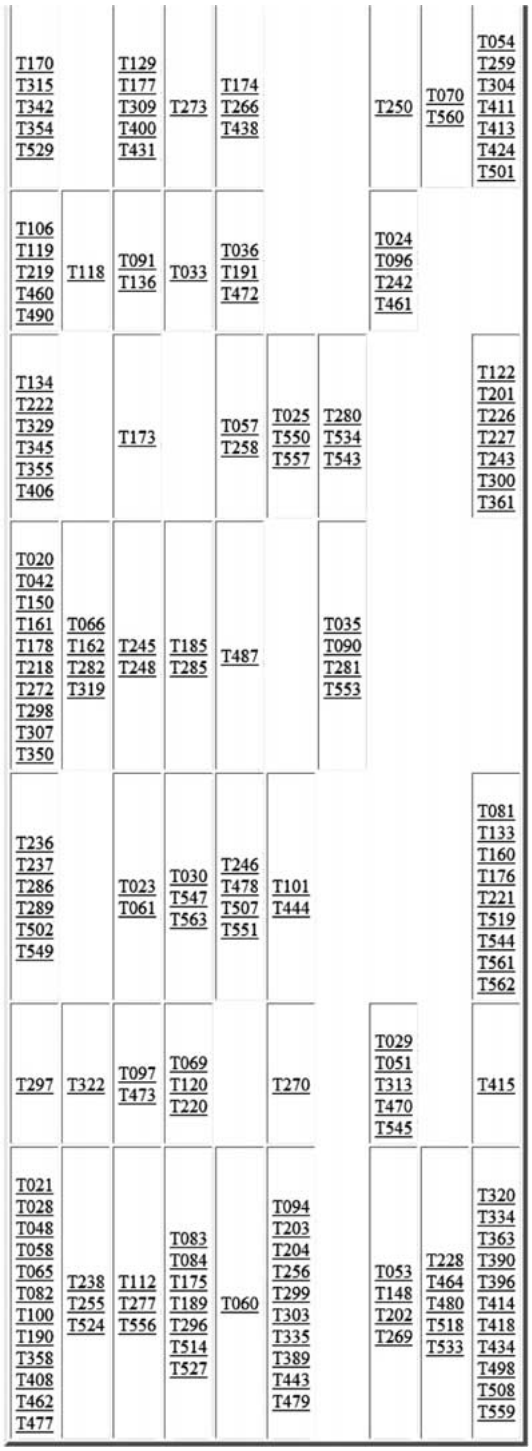

Figure 6. Document clusters on the lower half part of the 10×15 document map.

3. Performance of LSISOM

Our LSISOM method for dimensionality reduction using SOM based word category histograms leads to substantial economy in terms of processing resources and document representation. An important question, however, is whether the quality of clustering and document organization using our method is similar to that offered by the baseline method of clustering the original VSM data of 5923 dimensions using a self organizing map (we will refer to this method as standard SOM – SSOM approach). Given the unsupervised nature of the data, such a comparison of clustering results is not straightforward. In principle, a method based on correlations of membership matrices can be followed. For each of the two methods, a membership matrix of dimension 420×420 can be formed with each element $M(i, j)$ in the matrix representing the relationship between document i and document j. If the two documents belong to the same cluster in the clustering result, then $M(i, j) = 1$; otherwise $M(i, j) = 0$.

A measure of similarity between the maps obtained by SSOM and LSISOM would be the Pearson correlation coefficient between the membership matrix elements of the two methods:

$$\rho = \frac{\sum_{ij}(M_{\mathrm{SSOM}}(i,j) - \overline{M}_{\mathrm{SSOM}})(M_{\mathrm{LSISOM}}(i,j) - \overline{M}_{\mathrm{LSISOM}})}{\sqrt{\sum_{ij}(M_{\mathrm{SSOM}}(i,j) - \overline{M}_{\mathrm{SSOM}})^2}\sqrt{\sum_{ij}(M_{\mathrm{LSISOM}}(i,j) - \overline{M}_{\mathrm{LSISOM}})^2}} \tag{3}$$

Starting from different randomly chosen initial weight configurations, we have produced $N = 25$ different maps clustering the documents with SSOM and an equal number of maps clustering the documents with LSISOM and computed average correlation indices over the 25 runs. The average Pearson correlation coefficient between membership matrices for SSOM and LSISOM is equal to 0.401 ± 0.030 (average plus/minus one standard deviation is reported) and is therefore not very significantly positive. However, we also note that the average Pearson correlation coefficient between membership matrices using only the SSOM method starting from different initial weights is also quite low, equal to 0.468 ± 0.037. This indicates that SSOM is not consistent in producing steady clustering results. This phenomenon is not unusual with large scale clustering problems [19] and can be partially attributed to the existence of proximity ties among the input vectors [11]. In any case, this lack of consistency in the clustering result starting from different initial weights calls for a reevaluation of our procedure for estimating the similarity of SOMs resulting from the two methods, since there is effectively no stable SSOM map arrangement with which to compare maps obtained by the LSISOM approach.

To this end, we will now introduce the concept of 'steady pairs' of documents, i.e., pairs of documents that are assigned to the same cluster in all runs of the clustering algorithm starting from different initial weights. The following question arises: What is the proportion of steady pairs of documents obtained using

standard SOM on the original VSM data, that are also steady pairs using our LSISOM method? We have determined the set of steady pairs of documents for SSOM and LSISOM in 25 trials starting from different initial weights. Using these pairs, we have evaluated this proportion and find that 74.8% of the steady pairs obtained using SSOM continue to remain steady pairs using LSISOM. Moreover, 95.8% of the members of steady pairs obtained using SSOM have a distance of at most 1 (i.e., in the neighborhood with the 4 nearest neighbors) on the final map grid obtained using LSISOM. This percentage becomes 98.7% for members of steady pairs obtained using SSOM that remain in nodes separated by distances not greater than $\sqrt{2}$ on the map grid obtained using LSISOM (neighborhood with 8 nearest neighbors). Moreover, the number of steady pairs obtained using LSISOM (291 pairs in total) is larger than the number of steady pairs obtained using SSOM (159 pairs in total), meaning that the clustering effected by LSISOM is less prone to variation due to different initial conditions. This improvement in clustering stability and robustness is further supported by the evaluation of two more indices: First, the Pearson correlation coefficient between the membership matrix elements using LSISOM is 0.579 ± 0.038, significantly larger than the corresponding figure for SSOM. Secondly, we consider the Davies–Bouldin index [6] as a standard measure for the quality of clustering in terms of intra-cluster variability and inter-cluster separability, with lower Davies–Bouldin indices indicating better quality of clustering. The average ratio of the Davies–Bouldin index for maps obtained using LSISOM over the Davies–Bouldin index for maps obtained using SSOM is 0.780, indicating an improvement in clustering quality using our method.

A final question is whether this improvement in clustering quality and the resulting increase in the number of steady pairs of documents using our method also leads to thematically better clustering. The question that has to be asked is whether the expansion of some clusters due to the addition of extra steady pairs is thematically plausible, i.e., if the extra documents are indeed thematically related but have been actually missed by the original SOM clustering method.

For example, when analyzing the resulting steady pairs of the SSOM method in all $N = 25$ different maps we find that documents T024, T096, and T242, always form a single cluster. These documents deal with the relationship between India and Pakistan and the Kashmir conflict. An analysis of the steady pairs obtained with the LSISOM method for an equal number of maps indicates that the thematically related document T461 (entitled '*Pakistan - Whose Ally?*') is always added to the cluster formed by the previously mentioned documents which indicates that the expansion of the cluster is indeed valid.

Another example are documents T170, T342, and T354 forming consistently a single cluster with the SSOM method. These documents are related to the Profumo – Keeler scandal in Great Britain and to British politics. The titles of these documents are '*Great Britain – What Ever Happened to Christine Keeler?*', '*Great Britain - Goddess of the Gravel Pit*', and '*Great Britain – While the Prisoner Sketched*

in Jail', respectively. The set of steady pairs obtained with the LSISOM method reveals that two more documents, namely T315 (entitled '*Great Britain – The Price of Christine*') and T529 (entitled '*Great Britain – Less than a Pound*') are always added to the cluster formed by the above documents. These documents are obviously thematically related to documents T170, T342, and T354, thus justifying the consistent mapping of all 5 documents onto a single cluster by the LSISOM method.

As a final example we can consider news articles T029 and T545 related to the fighting in Vietnam. These documents form a steady pair with the SSOM method. The consistent cluster formed by the LSISOM method consists of documents T029, T545, T051, T269, and T313. The focus of all these documents is on troop movements and helicopter fights and missions in Vietnam thus expanding the SSOM steady pair and forming a thematically concentrated cluster. A more detailed description of the Times new article collection and of the clustering effected by SSOM can be found at 'http://www.ifs.tuwien.ac.at/~andi/somlib/data/time60/time-map10 × 15_labels.html'.

4. Conclusions and Future Directions

In this work, we have presented a novel methodology for a completely automatic and unsupervised full-text analysis of document collections using SOM. The method, called LSISOM, is similar to the WEBSOM method which is based on the utilization of word category histograms. In the LSISOM method these histograms are obtained from the SOM clustering of the LSI representations of document terms which enhances their semantic aspects. This results in efficient clustering of the words and in the automatic construction of a thesaurus onto the word category map which is valid within the context of the entire document collection. In addition the representation of the documents by their word category histograms results in vast dimensionality reduction of the original feature space and in the computationally efficient implementation of SOM training for the creation of an ordered map of the document space.

A potential drawback of the method arises from the computational cost of computing the LSI representations of the terms. However, due to the existence of numerical routines such as the power or the Lanczos method [2] for sparse data matrices, SVD can in many cases be efficiently computed. For a sparse mxn data matrix $\boldsymbol{A}$ with about c nonzero entries per column, the computational complexity of SVD is of order O(mcn) [21].

An interesting option would be the utilization of either RP or the composite RP/LSI method for the computation of term representations. The main advantage of such an approach is the computational simplicity of RP since the cost of projecting a sparse mxn data matrix $\boldsymbol{A}$ with about c nonzero entries per column, is of order O(ckn) [21]. These alternative options are currently under investigation and we hope that we will be able to report soon on the corresponding results.

References

1. Ampazis, N. and Perantonis, S. J.: Evaluation of dimensionality reduction techniques for SOM clustering of textual data, In: *Artificial Intelligence and Applications (AIA 2002)*, Malaga, Spain, 2001.
2. Berry, M. W.: Large scale singular value computations, *International Journal of Supercomputer Applications* **6**(1) (1992).
3. Berry, M. W., Dumais, S. T. and O'Brien, G. W.: Using linear algebra for intelligent information retrieval, *SIAM Review* **37**(4) (1995), 573–595.
4. Bingham, E. and Mannila, H.: Random projection in dimensionality reduction: applications to image and text data, In: *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, pp. 245–250, San Francisco, CA, USA, 2001.
5. Chen, H., Schuffels, C. and Orwig, R.: Internet categorization and search: A machine learning approach, *Journal of Visual Communication and Image Representation* **7**(1) (1996), 88–102, Special Issue on Digital Libraries.
6. Davies, D. L. and Bouldin, D.: A cluster separation measure, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **1** (1979), 224–227.
7. Deerwester, S., Dumais, S., Furnas, G., Landauer, T. and Harshman, R.: Indexing by latent semantic analysis, *Journal of the American Society for Information Science* **41**(6) (1990), 391–407.
8. Ding, C.: A similarity based probability model for latent semantic indexing, In: *ACM SIGIR Conference Proceedings* (1999).
9. Honkela, T., Kaski, S., Lagus, K. and Kohonen, T.: Newsgroup exploration with WEBSOM method and browsing interface, Technical Report No. A32, Espoo, Finland. 1996.
10. Honkela, T., Kaski, S., Lagus, K. and Kohonen, T.: WEBSOM–self-organizing maps of document collections, In: *Proceedings of WSOM'97, Workshop on Self-organizing Maps, Espoo, Finland, June 4–6*. pp. 310–315, Espoo, Finland, Helsinki University of Technology, Neural Networks Research Centre, 1997.
11. Jain, A. K. and Dubes, R. C.: *Algorithms for Clustering Data*. Prentice Hall, New Jersey, 1988.
12. Jing, Y. and Croft, W. B.: An association thesaurus for information retrieval, In: *Proceedings of RIAO-94, Fourth International Conference Recherche d'Information Assistee par Ordinateur*, pp. 146–160, New York, US, 1994.
13. Johnson, W. B. and Lindenstrauss, J.: Extensions of Lipshitz mapping into Hilbert space, *Contemp. Math.* **26** (1984), 189–206.
14. Kaski, S., Honkela, T., Lagus, K. and Kohonen, T.: Creating an order in digital libraries with self-organizing maps, In: *Proceedings of WCNN'96, World Congress on Neural Networks*, September 15–18, pp. 814–817, San Diego, California. Mahwah, NJ, Lawrence Erlbaum and INNS Press, 1996.
15. Kohonen, T.: *Self-organization and Associative Memory*, Springer-Verlag, N.Y, 3rd edition. 1989.
16. Kohonen, T., Kaski, S., Lagus, K., Salojrvi, J., Honkela, J., Paatero, V. and Saarela, A.: Self organization of a massive document collection, *IEEE Transactions on Neural Networks* **11**(3) (2000), 574–585. Special Issue on Neural Networks for Data Mining and Knowledge Discovery.
17. Lagus, K., Honkela, T., Kaski, S. and Kohonen, T.: Self-organizing maps of document collections: A new approach to interactive exploration, In: Simoudis, E., Han, J. and U. Fayyad (eds.), *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, AAAI Press, Menlo Park, California, pp.238–243, 1996a.

18. Lagus, K., Kaski, S., Honkela, T. and Kohonen, T.: Browsing digital libraries with the aid of self-organizing maps, In: *Proceedings of the Fifth International World Wide Web Conference WWW5*, May 6–10, pp. 71–79, Paris, France, Vol. Poster Proceedings. EPGL, 1996b.
19. MacCuish, J. D., N. C. and MacCuish, N. E.: A pattern recognition approach to understanding the multilayer perceptron, *J. Chemical Information and Computer Sciences* **41** (2001), 134–146.
20. Merkl, D. and Tjoa, A. M.: The representation of semantic similarity between documents by using maps: application of an artificial neural network to organize software libraries, In: *FID'94, General Assembly Conference and Congress of the International Federation for Information and Documentation*, 1994.
21. Papadimitriou, C. H., Raghavan, P., Tamaki, H. and Vempala, S.: Latent semantic indexing: a probabilistic analysis, *JCSS* **61**(2) (2000), 217–235.
22. Rauber, A. and Merkl, D.: Using self-organizing maps to organize document archives and to characterize subject matter: how to make a map tell the news of the world, In: *Database and Expert Systems Applications (DEXA 1999)*, pp. 302–311, 1999.
23. Ritter, H. and Kohonen, T.: Self-organizing semantic maps, *Biol. Cyb.* **61**(4) (1989), 241–254.
24. Salton, G. and McGill, M.J.: *Introduction to Modern Information Retrieval*, New York, 1983.
25. Scholtes, J. C.: Unsupervised learning and the information retrieval problem, In: *IJCNN'91, International Joint Conference on Neural Networks*, pp. 95–100, Singapore, 1991.
26. Vesanto, J., Himberg, J., Alhoniemi, E. and Parhankangas, J.: *SOM Toolbox for Matlab 5*. Report No. A57, Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland, 2000.
27. Lin, X., D. S. and Marchionini, G.: A self-organizing semantic map for information retrieval, In: *Fourteenth Annual International ACM/SIGIR Conference on R & D In Information Retrieval*, pp. 262–269, 1991.