Indexing Audio Documents by using Latent Semantic Analysis and SOM

Mikko Kurimo [a]

[a]IDIAP CP-592, Rue du Simplon 4, CH-1920 Martigny, Switzerland
Email: Mikko.Kurimo@idiap.ch

This paper describes an important application for state-of-art automatic speech recognition, natural language processing and information retrieval systems. Methods for enhancing the indexing of spoken documents by using latent semantic analysis and self-organizing maps are presented, motivated and tested. The idea is to extract extra information from the structure of the document collection and use it for more accurate indexing by generating new index terms and stochastic index weights. Indexing methods are evaluated for two broadcast news databases (one French and one English) using the average document perplexity defined in this paper and test queries analyzed by human experts.

1. INTRODUCTION

The development in large-vocabulary continuous speech recognition (LVCSR) has made it possible to automatically process big databases of recorded audio documents such as broadcast news, interviews etc. An important application of LVCSR is the automatic indexing of spoken audio by extracting index terms from the speech which is decoded by a speech recognizer. Although the words in the spoken documents cannot be recognized with 100 % accuracy, an automatically generated index can still be very useful for certain applications. For example, on radio and television there are huge amounts of broadcast data for which the collection and indexing without an excessive delay would require a prohibitive amount of human labour. Even if many important words are missed by the automatic speech decoding, the rest may provide enough information for the indexing system so that most of the relevant documents can be found for given queries. Efficient audio indexing is also highly relevant for, e.g., the multimedia industry and interactive TV, since it provides possibilities for easy to use consumer interfaces and on-line remote access into archived recordings.

The motivations for this paper are that we could enhance the audio indexing by extracting more information from the documents than just the most probable decoding and then by clustering the documents semantically. The clustering based on the most important semantical content reduces noise coming from the choice of words in the documents and recognition errors. Thus, a document can be indexed also for terms that were not found by the decoding, but which only appear in other semantically related documents and are probably relevant for the current document as well.

Self-Organizing Maps (SOMs) have successfully been applied to organize large text archives [16,14] by presenting the documents as smoothed histograms of the word categories that match with the document content. In this paper SOMs are used to cluster documents based on document vectors which are weighted averages of the vectors representing the words (or stems) decoded from the speech. The objective is to associate the documents with the index terms that describe well the main (latent) semantics of the documents and will rank the documents as well as possible according to the terms in a given document query.

2. INDEXING SPOKEN AUDIO

The work presented here is related to the THISL project (Thematic Indexing of Spoken Language) which is an ESPRIT Long Term Research project for speech retrieval [1]. The project aims to explore the limits of state-of-the-art LVCSR, IR (Information Retrieval) and NLP (Natural Language Processing) technologies for indexing and retrieval of television and radio data. The target application is a "news-on-demand" system which recalls the relevant parts of audio or video broadcasts based on a query from the user.

A prototype system for THISL has already been made for British and North American broadcast news based on the ABBOT [24] LVCSR system and a probabilistic IR system [1]. The system has been evaluated in the TREC-7 (Text Retrieval Conference) SDR track (Spoken Document Retrieval) [10,21]. Demonstrator systems have also been built for other English and French databases (e.g. [17]).

The basic approach for audio indexing can be divided into several consecutive phases:

1. The audio broadcast is recorded and preprocessed for speech recognition.
2. The recognizable speech is separated from music and other non-speech sounds.
3. Text files are created from the most probable decoding hypothesis.
4. The text files are indexed using the decoded words.
5. The queries are processed and relevant documents are retrieved.

The latest developments of the THISL broadcast news retrieval system are described in [1]. Corresponding full-text recognition based indexing approaches are currently used also by several other groups, e.g. [3,13]. Alternatively indexing can be based on keyword spotting or phone recognition [19]. The advantages of these systems are computationally lighter speech recognition and no out-of-vocabulary word problems. However, full-text recognition can constrain the task using pronunciation dictionaries and language models and thus provide a more robust text retrieval [1].

3. LATENT SEMANTIC ANALYSIS

Latent Semantic Analysis (LSA) [9] is used for modeling text data based on semantic structures found by analyzing the co-occurrence matrix of words and documents. These models project the data into lower dimensional subspaces by finding the most relevant structures. It is important that by focusing to the relevant structures in the data, the amount of noise originating, e.g., from speech recognition errors, is reduced as well. LSA is often associated with Principle Component Analysis (PCA) or Singular Value Decomposition (SVD) by which the LSA is normally generated. In document indexing LSA is applied to find out the essential index terms to which the documents should be associated.

LSA has traditionally been based on the idea that the data is efficiently compressed by extracting orthogonal components directed so that each new component minimizes the projection error remaining from previous components. For indexing, the document collections are usually presented as a matrix A where each column corresponds to one document and each row the existence of a certain word [25]. This representation looses the information about the word positions and groups in the document as it is mainly intended to determine only in which documents the words are used.

With SVD the word-document co-occurrence matrix is decomposed as $A = USV^T$ to find the singular values and vectors. By choosing the n largest singular values from S we obtain a reduced space where A is approximated by the estimate A_n [9]

$$A_n = U_n S_n V_n^T \; . \tag{1}$$

In this n-dimensional subspace the word w_i can be coded as

$$x_i = u_i S_n / \|u_i S_n\| \tag{2}$$

by using the normalized row i of matrix $U_n S_n$. We can then get smoothed representations by clustering the words or the documents using the semantic dissimilarity measure [5]

$$d(w_i, w_j) = x_i x_j^T \; . \tag{3}$$

In practice and especially in spoken documents, the documents are short and important words quite rare. To still get meaningful distributions of the index words in the models, a careful smoothing is needed [5]. This is generally done by clustering similar documents together and using the average document vector of each cluster to represent the cluster members. The cluster vectors will also generate a smoothed representation of the documents, since they integrate the content of several semantically close documents into one model. The clusters can be interpreted as automatically selected topics based on the given document collection.

To avoid quantization error between the document and its nearest cluster, a set of nearest clusters (or even all the clusters) can be used to compute the smoothed mapping. For example, we can consider their weighted average based on distance, so that nearby clusters will have the strongest effect. This generalization matches well the broadcast news example, since one section can be relevant to several topics.

4. USING SOM FOR LSA

The main contribution of this paper is the idea of using the SOM to compute a LSA based index for spoken documents in a way which is more suitable for very large data collections. With very large document collections like broadcast news, recorded over a long time, the dimensionality of matrix A (the word-document co-occurrence matrix) becomes too large to handle. However, the matrix is sparse, because only a small subset of the very large vocabulary is actually used in one document. There exist efficient methods to compute the SVD for sparse matrices such as the Single Vector Lanczos iteration [6] which lower the computational complexity significantly. However, it can still be difficult to always obtain an acceptable solution using this kind of iterative approximation methods.

By *Random Mapping* (RM) [22] we can artificially (randomly) and quickly generate approximately orthogonal vectors for the words and present the documents as an average vector of the words. In fact, because the co-occurrence matrix is usually very sparse, we can get quite a good approximation with a considerably lower computational complexity than with SVD, already with only 100 – 200 dimensional random vectors [15]. By using this approximation it becomes feasible to use a very large vocabulary and also to expand the index later by adding new documents and words.

For automatically decoded documents we must somehow take into account that documents are not completely described by the decoded words. Some relevant words are often lost or substituted by fully irrelevant ones. Clustering has the advantage of mapping the decoded documents based on their whole content and in that way minimizing the effect of incorrect individual terms. In classical clustering methods such as LBG (Linde-Buzo-Gray) and K-means each cluster vector is the average of vectors only in that particular cluster. This adapts the clusters well to the fine structure of the data, but can make the smoothing sometimes inefficient. The more training vectors affecting each cluster, the smoother is the representation, and the more will the clusters reflect the major structures of the data. If we do the *clustering by SOM*, each training vector affects at the same time all clusters around the best one, which makes it also easier to train large number of clusters [18]. As learning proceeds in a SOM the density of the cluster vectors starts eventually to reflect the density of the training vector space. This will provide the strongest smoothing on sparse of areas and the highest accuracy on dense areas. Like the RM document vectors, the SVD document vectors can be clustered by SOM as well to further reduce noise and gaining new index terms by mapping the documents to the clusters.

If we train the SOM into a two-dimensional grid, the automatic ordering will provide a visualization of the structures in the data (Figure 1). If the display is suitably labeled, we can see the dominant clusters and directions and get immediately a conception of the area where the chosen document lies [11,16]. For more thorough database exploration, a graphical interface, like WEBSOM [11], can be used to virtually move inside any point on the map and examine the document space around it.

5. EXPERIMENTS

5.1. Evaluation metrics

The correct evaluation of a spoken document index is a difficult task. Indexes prepared in a different way describe documents using the same or different index terms and, thus, might return different documents, for the same terms given as a query. In general, it is not possible to automatically judge which documents are relevant to a given query. For the user of the index it is also very important how the retrieved documents are ranked, i.e. the most relevant ones should be on the top. However, a proper comparison of the different ranking lists is even more difficult than just judging whether the results are relevant or not [10].

In this paper we apply the test used in the latest TREC evaluation for SDR track [10]. For a database of North American Business news a set of text decoding hypothesis

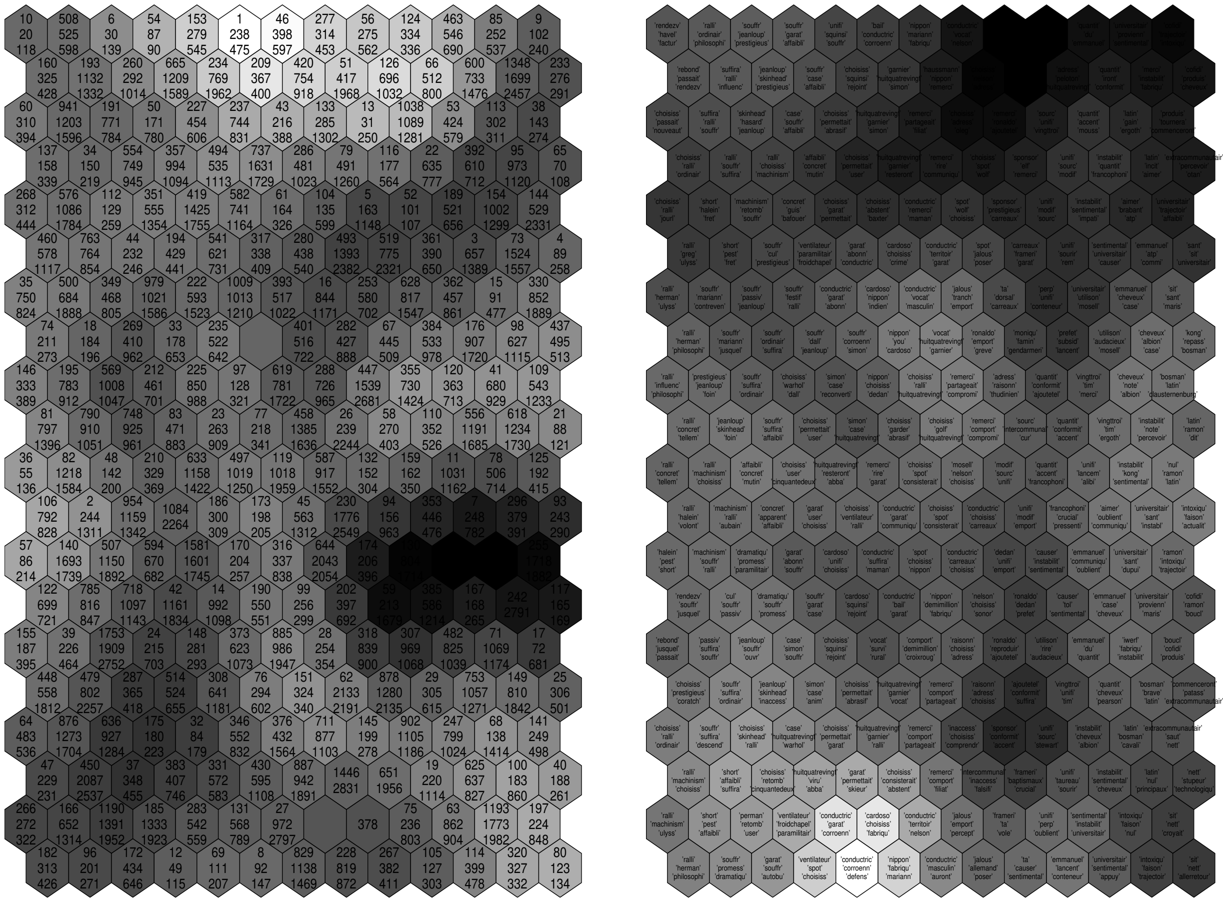

Figure 1. Examples of visualizing an indexed document collection. Each cell corresponds to one cluster (node) in the SOM grid. The vectors of neighboring cells are usually also near each other in the original high-dimensional vector space. The color of the cells is here used to show the distance between the cluster and the selected test document. A light color means a short distance. The numbers in cells are here (picture on the left) used to show the pointers to the documents that are closest to that cluster. Another way to study the clusters is to find the best matching index terms (a different database on the right).

using different speech recognizers was provided. TREC provided as well a set of carefully composed test queries and relevance judgments by human experts for the documents concerning each query. Several measures were defined to compare the relevance of the retrieved set of documents. The two most important used in this paper are the *recall*, which is the proportion of the relevant documents which are obtained, and the *precision*, which is the proportion of the obtained documents which are relevant. A meaningful comparison for ranked retrieval lists is then to check the precision at different levels of recall or, as in this paper, by computing the *average precision* (AP) over all relevant documents. In addition to AP, we use another related measure which is the *average R-precision* (RP) defined by the precision of the top R documents, where R is the total number of the relevant documents.

For the databases where no relevance judgments are available, we propose a new concept called the *average document perplexity* [17] to give a numerical measure of how well an index describes the documents. In speech recognition the measure of perplexity is commonly used to quantify the relative difficulty of a recognition task. The perplexity is a measure of the strength or predictive power of the LM (Language Model) constraints and it is also widely used to compare LMs when it is too expensive to compute every time the actual WER (Word Error Rate) for whole speech recognition system [8]. The perplexity for the words $w_1, \ldots, w_T$ in the test set can be defined as

$$PP = \exp(-\frac{1}{T}\sum_{i=1}^{T} \ln \Pr(w_i|\mathrm{LM}))\,. \tag{4}$$

For document set models the perplexity can be defined using the vector space representation of words and documents so that instead of $\Pr(w_i|\mathrm{LM})$s we have the probabilities given by the LSA model for the test document. The LSA probabilities are computed using the normalized matches between the vectors of the index terms (words or stems) and the vector of the test document (or its smoothed version). A high word match means that the word is very likely to exist in the test document and the more unlikely words there are in the test document, the higher the perplexity. Thus a higher average document perplexity means also that the models have less predictive power for the tested documents and the index might be worse. However, perplexity is by no means a substitute for the actual retrieval test and, as it is well known from speech recognition experiments, even significant improvements in perplexity do not necessarily imply improvements in the actual WER [12].

5.2. Tested indexing methods

After the tested news databases were processed as explained in Section 2, the obtained text files were used to prepare the indexes. Since full lattice decoding results were not yet available, the indexing was made based on the most probable decoding only. The French LVCSR system based on a hybrid HMM/MLP model that was used to decode the databases is described in [7] with latest details in [4,17].

The index that was called "default THISL", according to the first THISL prototype version [2], creates an inverted file using the stems of the decoded words directly as the index terms. The inverted file is basically a list of words with pointers to relevant documents. The stemming was made using the Porter stemming algorithm [20], so that the stop words were first filtered out and then the suffices removed from the rest of the words to get the stems. The stemming algorithm is tuned only for English so that the French stems are probably not optimal. The stop list is an edited list from the most frequent words in the language.

The LSA indexes were made by first preparing the smoothed document vectors as explained in Sections 3 and 4. For the traditional SVD approach sparse SVD with 125 first singular values and vectors was computed and the normalized word codes (Equation 2) of the word stems was used to form the document vectors. The RM + SOM approach was based on 200-dimensional normalized random vectors for the stems and a two-dimensional SOM of 260 units for the document vectors. A SOM of the same size was also used for smoothing the SVD based document vectors.

For the construction of the document vectors, an importance weighting was used for the word stems in both the RM + SOM and the traditional LSA (unlike in [17]). The rarer the word is in the collection, the better it usually describes and discriminates the documents. Thus, the importance weight reflects the relevance of a word to the whole document collection and it can be derived, e.g., using the mutual information (defined with entropy) [26] or its simpler approximation, the Inverse Document Frequency (IDF) [23]. The forms of IDF used here scaled within [0,1] are the simple

$$\mathrm{IDF}_i = 1/n_i \, , \tag{5}$$

and the logarithmic

$$\mathrm{IDF}'_i = 1 - \log n_i / \log \max n_i \, , \tag{6}$$

where n_i is the number of documents where the stem i exists [23].

To determine the best index terms for each document the smoothed document vectors are compared to all the stem vectors. The indexing was made stochastically so that the index words were weighted by the LSA scores scaled within [0,1]. To integrate the LSA index with the basic index, the index terms selected directly from the actual decoding were added with weight 1.0. Since it was not feasible to index every document with all index terms the limit of significance was determined by assuming LSA scores normally distributed and selecting all the terms corresponding to scores above the 99 % significance level.

The LSA scores of a document computed for all the index terms, actually approximate the probability:

$$\Pr(\mathrm{doc}|\mathrm{word}) = \Pr(\mathrm{word}|\mathrm{doc}) \Pr(\mathrm{doc}) / \Pr(\mathrm{word}) \, , \tag{7}$$

where the probability of each word Pr(word|doc) can be computed smoothed by the K (best-matching) clusters $C_1, \ldots, C_K$ weighted by their similarity with the current document

$$\Pr(\mathrm{word}|\mathrm{doc}) = \sum_{k=1}^{K} \Pr(\mathrm{word}|C_k) \Pr(C_k|\mathrm{doc}) \, . \tag{8}$$

After the LSA index is made, it can be used similarly as the "default THISL" index [21]. Queries are processed by eliminating stop words and mapping other words into their stems. To find the best matches, the documents are scored based on the number of matches between the query terms and the document using the index. The scores are normalized using weights for document length and the term frequency in the collection [23].

6. RESULTS

Results are given here for two broadcast databases. The first database has French speaking news and in the decoding used here the WER was high and varied a lot between different sections. The average perplexity results in Table 1 indicate that the more smoothing is applied, the higher is the perplexity on the training data. (Smaller neighborhood and larger number of SOM units imply less smoothing). The perplexities between RM and SVD based indexes are not directly comparable. Since no test queries were yet available for this database, another better standardized test set was also analyzed (Table 2 and 3).

Table 1
Average document perplexities (PP) for the French database. SOM0 is SOM trained with 0-neighborhood (equivalent to an on-line adaptive version of the classical K-means clustering) and SOMb a larger SOM (600 units). For clustered systems (SOM) the smoothed model is made using the weighted average of 10 best-matching clusters (as explained in Section 2). For the non-clustered methods (RM, SVD) the weighted average of 20 best-matching actual document vectors is used for smoothing. When no clustering was used independent test data could be simulated by ignoring the current document to give perplexities 1.94 and 2.46 for RM and SVD, respectively.

Index	PP
RM	1.68
RMSOM0	1.75
RMSOM	1.85
RMSOMb	1.80
SVD	2.14
SVDSOM0	2.47
SVDSOM	2.62
SVDSOMb	2.33

Table 2 presents perplexities and test query evaluations for the TREC test set. The speech decoding used here had a 36 % average WER. More results (and using another decoder) have been presented in [17]. The query expansion, where not only the index terms related to the query are checked, but also terms that are commonly associated with them in reference databases [27,1], was not used here. From Table 2 we see that the average precision improves with SVD and even further when we smooth the models by SOM. The closer comparison in Table 3 shows, e.g. that LSA retrieves many more documents than the references, including also slightly more of the relevant ones. By looking at the lowest standard recall level 0.10, which gives the precision of the highest ranked documents, LSA seems also to do quite well. For higher recall levels the precision of LSA drops below that of the baseline, because the cost of the higher total recall seems to be a vast increase of irrelevant documents. In Table 2 the document perplexity for RM index decreases as stronger smoothing is applied, but the AP and RP indicators do not show any clear improvement. For SVD coding the AP and RP indicators show improvements with smoothing, but the perplexity does not change much.

Table 2
Evaluation results for the TREC test set. AP is the average precision, RP the R-precision. "THISL default" is a baseline index without LSA and "perfect" is an index based on the correct transcriptions. As in Table 1, the simulated test data perplexity gave 2.7 and 1.8 for the non-clustered RM and SVD, respectively.

Index	AP	RP	PP
RM	0.33	0.34	2.6
RMSOM0	0.33	0.35	2.2
RMSOM	0.34	0.36	2.1
SVD	0.35	0.34	1.7
SVDSOM0	0.37	0.34	1.8
SVDSOM	0.38	0.34	1.8
THISL default	0.37	0.37	
"perfect"	0.43	0.41	

Table 3
Some finer details of the comparison between the reference systems and the best LSA system (SVDSOM) for Table 2. "ranked" is the average number of documents ranked per query, "recall" the total recall, and "P.10" the precision at recall level 0.10.

	"perfect"	S1 decoding	
	decoding	ref.	LSA
ranked	0.29	0.31	0.66
recall	0.92	0.91	0.96
P.10	0.65	0.62	0.65
AP	0.43	0.37	0.38
RP	0.41	0.37	0.34

7. CONCLUSIONS

This paper describes a system for decoding spoken documents and indexing them based on the latent semantic analysis of the document contents. A new computationally simple approximative approach is suggested for LSA in large document collections. To smooth the LSA models we apply clustering with a SOM. This provides as well an organized view over the contents of the document collection. Experiments are made using French and American news databases and for the latter we provide the results of relevance judgments using standardized test queries. To measure the predictive power of the models we define a new document perplexity measure.

The results show that the proposed way to construct LSA index by RM + SOM does not give quite as accurate retrieval results (AP) as the SVD based LSA or the baseline THISL index. At a higher recall level (RP) the precision of RM-based indexes is between that of SVD and the baseline THISL. However, at the lowest recall level (P.10), which is probably the most useful for the interface users, the precision provided by SVD+SOM was the highest and as good as by the "perfect" index.

From a computational point of view the RM + SOM is better than SVD, since it is much faster and there are much less complexity problems as the number of documents and words increases. It is also convenient that we do not need to change the old document vectors as the database is updated. The clustering of models is favorable, since the indexing is faster with smaller total number of models and smaller number of selected best models. The SOM algorithm behaves well for large document collections, because it is not affected by the vocabulary size and only almost linearly by the number of documents as opposed to typical SVD methods where the complexity is usually much higher.

For further research we have left the integration of acoustic confidence measures and n-best hypothesis into the presented stochastic index, and the testing of the query expansion method with the LSA index. For the French databases the same stemming algorithm as for English has so far been used, but because the suffixes are different, we will probably have to implement a totally new algorithm. Further development of the ranking strategies might be useful for LSA, since we get significantly more matching documents and there is also more useful information included in the indexing weights. Another interesting aspect is the use of data visualization to help understand the structures in the database and to use suitable words in queries.

ACKNOWLEDGMENTS

This work was supported by ESPRIT Long Term Research Project THISL. I wish to thank Dr. Chafic Mokbel and the speech group in IDIAP for useful discussions concerning the methodology and comments concerning to this paper.

REFERENCES

1. Dave Abberley, David Kirby, Steve Renals, and Tony Robinson. The THISL broadcast news retrieval system. In *ESCA ETRW workshop on Accessing Information in Spoken Audio*, Cambridge, UK, April 1999.
2. Dave Abberley, Steve Renals, and Gary Cook. Retrieval of broadcast news documents

with the THISL system. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3781–3784, 1998.

3. J. Allan, J. Callan, W.B. Croft, L. Ballesteros, D. Byrd, R. Swan, and j. Xu. INQUERY does battle with TREC-6. In *Proceedings of the Sixth Text Retrieval Conference (TREC-6)*, pages 169–206, 1998.
4. Johan Andersen. Baseline system for hybrid speech recognition on french. COM 98-7, IDIAP, 1998.
5. Jerome R. Bellegarda. A statistical language modeling approach integrating local and global constraints. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 262–269, 1997.
6. Michael W. Berry. Large-scale sparse singular value computations. *Int. J. Supercomp. Appl.*, 6(1):13–49, 1992.
7. Herve Bourlard and Nelson Morgan. *Connectionist Speech Recognition - A Hybrid Approach*. Kluwer Academic Publishers, 1994.
8. Stanley F. Chen, Douglas Beeferman, and Ronald Rosenfeld. Evaluation metrics for language models. In *DARPA Broadcast News Transcription and Understanding Workshop*, 1998.
9. S. Deerwester, S. Dumais, G. Furdas, and K. Landauer. Indexing by latent semantic analysis. *J. Amer. Soc. Inform. Sci.*, 41:391–407, 1990.
10. John S. Garofolo, Ellen M. Voorhees, Cedric G. P. Auzanne, and Vincent M. Stanford. Spoken document retrieval: 1998 evaluation and investigation of new metrics. In *ESCA ETRW workshop on Accessing Information in Spoken Audio*, 1999.
11. Timo Honkela, Samuel Kaski, Krista Lagus, and Teuvo Kohonen. Newsgroup exploration with WEBSOM method and browsing interface. Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland, 1996.
12. R. Iyer, M. Ostendorf, and M. Meteer. Analyzing and predicting language model improvements. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 254–261, 1997.
13. S.E. Johnson, P. Jourlin, G.L. Moore, K. Sparck Jones, and P.C. Woodland. The Cambridge university spoken document retrieval system. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 49–52, 1999.
14. Samuel Kaski, Timo Honkela, Krista Lagus, and Teuvo Kohonen. WEBSOM - self-organizing maps of document collections. *Neurocomputing*, 21:101–117, 1998.
15. Samuel Kaski. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, volume I, pages 413–418, 1998.
16. Teuvo Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1997. 2nd extended ed.
17. Mikko Kurimo and Chafic Mokbel. Latent semantic indexing by self-organizing map. In *ESCA ETRW workshop on Accessing Information in Spoken Audio*, Cambridge, UK, April 1999.
18. Mikko Kurimo. *Using Self-Organizing Maps and Learning Vector Quantization for Mixture Density Hidden Markov Models*. PhD thesis, Helsinki University of Technology, Espoo, Finland, 1997.

19. Kenney Ng and Victor W. Zue. Phonetic recognition for spoken document retrieval. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 325–328, 1998.
20. M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, July 1980.
21. Steve Renals, Dave Abberley, Gary Cook, and Tony Robinson. THISL spoken document retrieval. In *Proceedings of the Seventh Text Retrieval Conference (TREC-7)*, 1998.
22. Helge Ritter and Teuvo Kohonen. Self-organizing semantic maps. *Biol. Cyb.*, 61(4):241–254, 1989.
23. S.E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *J. Amer. Soc. Inform. Sci.*, 27(3):129–146, 1976.
24. T. Robinson, M. Hochberg, and S. Renals. The use of recurrent networks in continuous speech recognition. In C. H. Lee, K. K. Paliwal, and F. K. Soong, editors, *Automatic Speech and Speaker Recognition - Advanced Topics*, chapter 10, pages 233–258. Kluwer Academic Publishers, 1996.
25. G. Salton. *The SMART Retrieval System-Experiments in Automatic Document Processing*. Prentice-Hall, NJ, 1971.
26. Matthew Siegler and Michael Witbrock. Improving the suitability of imperfect transcriptions for information retrieval from spoken documents. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 505–508, 1999.
27. J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proc. ACM SIGIR*, 1996.