Jonathan L. Elsas. An Evaluation of Projection Techniques for Document Clustering: Latent Semantic Analysis and Independent Component Analysis. A Master's Paper for the M.S. in I.S degree. July, 2005. 34 pages. Advisor: Robert M. Losee

Dimensionality reduction in the bag-of-words vector space document representation model has been widely studied for the purposes of improving accuracy and reducing computational load of document retrieval tasks. These techniques, however, have not been studied to the same degree with regard to document clustering tasks. This study evaluates the effectiveness of two popular dimensionality reduction techniques for clustering, and their effect on discovering accurate and understandable topical groupings of documents. The two techniques studied are Latent Semantic Analysis and Independent Component Analysis, each of which have been shown to be effective in the past for retrieval purposes.

Headings:

Information Retrieval
Statistical Methods/Evaluation

AN EVALUATION OF PROJECTION TECHNIQUES FOR DOCUMENT CLUSTERING: LATENT SEMANTIC ANALYSIS AND INDEPENDENT COMPONENT ANALYSIS.

by
Jonathan L. Elsas

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

July 2005

Approved by

_____

Robert M. Losee

INTRODUCTION

Document clustering has long been an important problem in text processing systems, dating back to Salton's SMART information retrieval system (Salton & McGill, 1983), and recently becoming popular in internet search engines such as Vivisimo (http://vivisimo.com) and MetaCrawler (http://www.metacrawler.com/). The goal in most of these document clustering systems is to automatically discover, in the absence of metadata or a pre-existing categorization, sensible topical organizations of the documents. This clustering task, like many text processing tasks, is made difficult by the extremely sparse and high-dimensional nature of text data. For this reason, dimensionality reduction and term-space projection techniques have been proposed to alleviate this *curse of dimensionality* and make a variety of automatic text processing tasks more tractable. Dimensionality reduction techniques have been studied in great depth in document retrieval systems (Bingham & Mannila, 2001; Deerwester *et al.*, 1990; Efron, 2002; Isbell & Viola, 1999), but there is a dearth of information on how dimensionality reduction relates to document clustering. This study will begin to address this question by empirically evaluating the effect on document clustering of two popular dimensionality reduction techniques: latent semantic analysis and independent component analysis.

DOCUMENT CLUSTERING

Document clustering has been used widely in text processing systems. The goal of document clustering is to identify groups of documents, or clusters, so that documents within a cluster are similar and documents in different clusters are dissimilar. This goal is commonly phrased as maximizing intra-cluster document similarity while minimizing inter-cluster similarity (Zhao & Karypis, 2001). The organization of the clusters from a clustering algorithm can take several different forms: hierarchical clustering produces clusters that are related to each other in a tree-like hierarchy; probabilistic clustering produces a "soft" partitioning of documents, where each document has a probability of belonging to each cluster; and partitioning clustering produces mutually exclusive sets of documents. Many clustering algorithms have been proposed, the most popular being k-means (partitioning), expectation maximization (EM, probabilistic) and complete link (hierarchical). The reader is directed to (Jain *et al.*, 1999) for a review of data clustering methods and a variety of applications. The current study will limit its scope to probabilistic and partitional algorithms.

Document clustering was initially proposed to improve information retrieval performance in systems such as Salton's SMART system (Salton & McGill, 1983). In this context, it is assumed that documents relevant to a given query are more likely to be organized in the same cluster rather than different clusters. This assumption is known as the *cluster hypothesis* (van Risjbergen, 1979). Typically in retrieval systems using clustering as a performance enhancing tool, documents are clustered offline at indexing time. Then,

when a query is received in the system, the best cluster (or clusters) must be chosen based on the query, and then relevant documents are retrieved from within that cluster.

In retrieval applications such as these, clustering is used solely as a back-end preprocessing step to improve retrieval performance. Clustering has also been used as a tool to organize large document collections, attempting to automatically provide a sensible grouping of the documents. In this realm, clustering is often referred to as analogous to the table of contents in a book, whereas standard retrieval systems are analogous to a book's index (Dhillon & Modha, 2001). Towards this goal, document clustering systems must reflect the *topical* organizations of the documents in the collection, rather than the more abstract idea of documents being mutually relevant to the same query. These techniques have been used to organize large sets of retrieval results (Hearst & Pedersen, 1996; Zamir *et al.*, 1997) and document collections as a whole outside of the retrieval context (Cutting *et al.*, 1992; Efron *et al.*, 2004).

Some of the first work to take document clustering beyond a tool to improve retrieval performance was done in the Scatter/Gather system (Cutting et al., 1992). In this system a document collection was clustered, summarized and presented directly to the user as a tool to facilitate the interactive browsing of the collection. The system allowed users to iteratively select the clusters they are most interested in, re-cluster the documents in those clusters, and generate more fine organizations of the documents. Several other systems (Hearst & Pedersen, 1996; Zamir et al., 1997) have used clustering to organize the presentation retrieval results to the user in an analogous way, allowing the user to browse

interesting clusters instead of a ranked list of retrieval results. This type of presentation has been shown to improve retrieval performance (Leuski, 2001), and the presentation of clustered retrieval results has recently become a popular feature on large-scale web search engines such as Vivisimo (http://vivisimo.com) and MetaCrawler (http://metacrawler.com).

In addition to clustering for document organization, whether for retrieval or browsing, clustering has also been applied to document collections for the purposes of discovering latent factors. Similar to LSI, clustering for factor discovery aims to simultaneously reduce the dimensionality of the document representations and uncover hidden "concepts" or "topics" in the document collection. Instead of representing documents by vectors of term weights, in this model documents are represented by linear combinations of topic-clusters representing the document's strength of association to those topics. Noteworthy applications of clustering for latent factor discovery include Probabilistic Latent Semantic Analysis (Hofmann, 1999) and concept decomposition (Dhillon & Modha, 2001). In both of these methods, documents are represented by their degree of association with a small number of latent variables, and these latent variables are derived through a process of document-clustering.

A brief overview of clustering algorithms

An enormous variety of clustering algorithms have been applied to text data. Two of the most popular partitioning algorithms are k-means (Dhillon & Modha, 2001; Sinka &

Corne, 2002; Steinback *et al.*, 2000) and expectation maximization (EM) (Dasgupta, 1999, 2000; Hofmann, 1999). K-means creates a hard partitioning of the document collection by alternating between assignment of documents to the nearest cluster center (or centroid), and re-computing those centroids based on the newly assigned documents. The EM algorithm is an algorithm for probabilistically discovering an hidden, unobserved categorical variable; but is often used as a soft partitioning algorithm that creates a probabilistic clustering of documents. When used for clustering, the hidden variable is interpreted as defining the cluster membership. This is done by estimating a normal distribution (either multiple univariate normals or a single multivariate normal) for each category of the variable, or cluster, and assigning probabilities based on the likelihood of generating a particular document relative to each category's, or cluster's, distribution. Similar to k-means, this algorithm alternates between two states: an *expectation* step in which each document is assigned to a distribution based on the highest probability, and a *maximization* step in which the parameter estimates for each distribution are updated to that maximize the model likelihood based on the assigned documents. Both of these algorithms are typically initialized with a random configuration and converge when the either stability is achieved, in the case of k-means, or the likelihood function ceases to increase, in the case of EM. They are both guaranteed to converge at a solution that is at least locally optimal (Hastie *et al.*, 2001). For a more thorough review of clustering algorithms and some applications to text data, the reader is referred to (Ghosh & Ye, 2003) and (Jain et al., 1999).

The EM algorithm will be used in the experiments presented in this paper, and further exploration of some limitation of this algorithm will be elaborated on below.

NOISE, DIMENSIONALITY REDUCTION & LATENT FACTORS IN TEXT

A widely held tenet in the text processing literature is that textual data is extremely noisy. This noisiness is often thought of taking two forms: polysemy, or a single word having multiple meanings; synonymy, or multiple words having equivalent meanings. Additionally, the richness of human language and the variety of writing styles inevitably lead to imprecision when translating unstructured text into a highly structured machine understandable representation. This noise can be thought of as sampling error in the data: an underlying meaning is present, however through word choice and other stylistic factors some degree of random error is introduced into the document representations.

This noisiness coupled with the extreme high dimensionality of the bag-of-words vector-space document representation scheme poses many challenges to automatic text processing systems. In particular, clustering algorithms such as EM and k-means not only suffer from extended running times, but also frequently over-fit noisy high-dimensional data. As stated above, the EM algorithm converges to some locally optimal solution, and as the number of dimensions increase the number of locally optimal solutions expand dramatically. Some dimensionality reduction techniques can effectively overcome these problems by both reducing the computational requirements and

transforming the data in such a way that clusters are easier to discover (Dasgupta, 1999, 2000).

Many dimensionality reduction techniques have been proposed for text processing systems. Several, such as using stop-word lists and term frequency cutoffs rely solely on culling terms from the indexing vocabulary. Linear algebraic techniques, such as *projections*, have also been used extensively. A projection is a method of reducing the dimensionality of a data matrix through matrix multiplication, or linearly transforming the existing data matrix. Projections can arbitrarily reduce the dimensionality of any data matrix. A common way to visualize a simple projection is to envision a shadow cast by a three-dimensional object onto a flat surface. In this case, the three dimensions of the original object are projected onto two dimensions of the flat surface.

Among the recent favorites for term-space projections are Latent Semantic Analysis (LSA) and Independent Component Analysis (ICA). These techniques rely on statistical and linear algebraic techniques to project the document-term matrix onto a matrix of lower dimensions. In addition to reducing the dimensionality, it has been claimed that these techniques reduce or eliminate noise in the data, thereby revealing a level of latent semantic structure inaccessible in the original document representations (Deerwester et al., 1990; Isbell & Viola, 1999; Kolenda & Hansen, 1999). This latent semantic structure is discovered by utilizing the term and document co-occurrence structure, and discovering “interesting” dimensions in the data. These latent factors revealed in the data

are often thought of as approximating human-like knowledge or modeling topical concepts.

Latent Semantic Analysis

Latent Semantic Analysis (LSA) is the application of a linear algebraic technique, the singular value decomposition (SVD), to the document term matrix. Commonly in the document retrieval context, LSA is referred to as Latent Semantic Indexing (LSI), however the mathematical techniques behind both LSA and LSI are identical in general. The SVD is a matrix factorization given by (Deerwester et al., 1990; Strang, 1993)

$$A = U \sum V^T,$$

where A is the original data matrix, U and V are orthonormal matrices containing the left and right singular vectors of A and $\sum$ is a diagonal matrix containing the singular values of A. From the SVD of a matrix, a lower-dimensional matrix approximation can be derived by truncating the columns of the matrices U, V and both the rows and columns of $\sum$ to the first p (largest) singular values:

$$A \approx A_P = U_P \sum_P V_P^T.$$

The magnitude of a singular value directly correlates to the amount of variance explained by the corresponding singular vector. An assumption is made that the smallest singular

vectors represent random sampling error and can therefore safely be eliminated. The matrix approximation using the SVD is the best approximation of the original matrix in the least-squares sense and retains the maximum amount of variance from the original matrix (Strang, 1993).

In order to operate on the dimension-reduced version of A, a matrix projection must be derived that projects the full matrix A into the left singular values, U_P. This projection is given by

$$V_P \sum_P^{-1},$$

and when applied to A, gives the p-column truncation of U. (This projection assumes the documents are on the rows of A and terms are on the columns. If the converse is the case, this projection can be reversed to give the p-column truncation of V.)

In the vector-space information retrieval context, LSA has been widely researched (Deerwester et al., 1990; Efron, 2002), although results have been mixed and this technique is no longer considered state of the art. When using LSA for vector-space retrieval, the projection matrix above is applied to both the query vector and the documents in the collection, and similarity is measured in the reduced p-dimensional space. Analogously, in the context of document clustering, a clustering algorithm can be applied to the dimension-reduced document representation rather than the full-rank representation. This idea was explored in (Schutze & Silverstein, 1997), which found

very little difference in clustering effectiveness between an LSI-based projection and a simple method of truncating the document vectors.

Independent Component Analysis

Independent Component Analysis (ICA) is a *projection pursuit* technique originally developed for separation of mixed signals from unseen sources (Hyvarinen *et al.*, 2001). Projection pursuit is the seeking of dimensions in data that are "interesting" in that they exhibit some sort of structure not visible in the original data. LSI can be thought of a form of projection pursuit if interestingness is defined as the dimensions that capture the maximal variance. However, maximal variance is not typically considered an interesting aspect of the data; more often it is measured as conforming to some structural pattern or diverging from a given distribution (Hand *et al.*, 2001). ICA defines interestingness in terms of the directions that are statistically independent and least normally distributed. Departure from a normal distribution in this case judged in terms of the information-theoretic measure of entropy: a normal (Gaussian) distribution has maximal entropy and entropy decreases as the distribution of the data departs further from a normal distribution. Entropy is often thought of as a measure of randomness or lack of organization. A highly random or noisy distribution of data has a high entropy, and less random distribution that exhibits some structure or organization has lower entropy. Therefore by seeking out minimally entropic dimensions in the data, the most highly structured dimensions can be identified.

In ICA, the original data matrix A is decomposed into a *mixing matrix*, M, and a matrix of *sources*, X, representing the independent components:

$$A = X M.$$

And the independent components can be retrieved by inverting M thus creating an *un-mixing matrix* to recover the independent components:

$$X = A M^{-1}.$$

Typically, the data matrix A is "whitened", or transformed so that the columns are uncorrelated. This can be done with SVD along with dimensionality reduction, explained above:

$$X = A V_P \sum_P^{-1} M^{-1} .$$

And thus the projection matrix for ICA is:

$$V_P \sum_P^{-1} M^{-1}.$$

Just as with LSA, the original documents can be projected with this matrix and operated on in this space spanned by the independent components

When used with text processing, the independent components discovered in ICA are often interpreted as meaningful concepts (Isbell & Viola, 1999; Kolenda & Hansen, 1999). This offers an distinct advantage over LSA where the dimensions have so far eluded human interpretation. Although this may not be an essential feature when these projections are used in the back-end of a text processing system, it nonetheless makes ICA an appealing technique for dimensionality reduction and latent topic discovery.

ICA has been applied to text data in a variety of ways and yields some intuitively appealing results. This technique has been used in analogous ways to LSA in document retrieval (Isbell & Viola, 1999), for topic discovery in temporal text (Bingham *et al.*, 2003), and unsupervised identification of linguistic features such as parts of speech (Honkela & Hyvarinen, 2003).

METHODOLOGY

The experiments presented in this paper aim to compare the effectiveness of two different document projection techniques when used as a preprocessing step for topical segmentation of a document collection through clustering. The data used for these experiments comes from a real-world collection of web documents from several online directories.

The Web dataset contains 11,000 documents, with 1000 documents in each of 11 mutually exclusive categories (Sinka & Corne, 2002). Each category in this dataset is

associated with one of four broader themes, creating a shallow hierarchy of topics. Some of these categories are "Commercial Banks" and "Insurance Agencies" in the "Banking & Finance" theme and "Java" and "Visual Basic" in the "Programming Languages" theme. This hierarchical organization allows for the selection of documents in closely related categories such as "Java" and "Visual Basic", or documents in unrelated topics such as "Java" and "Commercial Banks". In this study several subsets of this collection will be used in order to approximate a easier and harder topical segmentation tasks. These subsets are WEB-JC including the documents from the Java and Commercial Banks classes and WEB-JV including documents from the Java and Visual Basic classes.

As in every document processing system, many decisions must be made during document preprocessing and conversion from unstructured text data to a numerical representation. In these experiments, the following decisions were made when building the bag-of-words document representations: all non-alpha characters were discarded; word boundaries were considered to be any white-space or punctuation; all words were converted to lower case and stemmed using Porter's stemmer (Porter, 1980); stop-words were removed using a standard IR stop list[1]; and infrequent terms that occurred in less than five documents were removed. A standard bag-of-words document representation scheme will be used, with term frequency (TF) term weighting. Previous studies have shown that no benefit is realized by using alternate weighting schemes such as TF-IDF in clustering experiments (Schutze & Silverstein, 1997; Sinka & Corne, 2002).

[1] Stop-words available at: http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/

The choice of clustering algorithm is also a critical component in the experimental setup, and many clustering algorithms have been applied to text data. For these experiments the expectation maximization (EM) algorithm will be used, which estimates a normal distribution on each dimension for each cluster. This method has been explored extensively for document clustering (Dasgupta, 1999, 2000) and been used with text for latent factor discovery (Hofmann, 1999). Rather than a strict partitioning of documents, the output of the EM algorithm is a soft, probabilistic clustering where each document has a non-zero probability of membership to each cluster. In preliminary experiments, for the majority of documents the EM algorithm assigned a near-one probability to one cluster and a near-zero probability to all other clusters, thus essentially creating a hard partitioning. For this reason, the results reported below will treat the EM output as a partitioning rather than a soft clustering: each document is assigned to the one cluster with the highest probability of membership. This simplifying assumption, in addition to reflecting the tendency of EM to assign extreme probabilities, makes comparison to existing (hard) document classifications more straightforward.

The EM algorithm, like many partitional clustering algorithms, is typically initialized with a random parameterization and terminates when a local maxima is found. It is possible to select a poor initial configuration, and thus find a sub-optimal local maxima. In order to mitigate this risk, this study will follow the approach taken in (Zhao & Karypis, 2001); ten clustering solutions will be built for each configuration and the best solution, the one with the highest log-likelihood, will be retained.

As stated above, the goal of this study is to evaluate two dimensionality reduction (or term-space projection) techniques for use in document clustering. A series of experiments will be conducted to host this comparison, evaluating the techniques across several different parameterizations. For the three document collections described above (WEB-JC and WEB-JV), the dimensionality will be reduced to 10, 50, 100 and 150 dimensions, and each of these reduced-dimensionality matrices will be clustered into 5, 10, 15 and 20 clusters as in (Zhao & Karypis, 2001). This yields $2 \times 4 \times 4 \times 10 = 320$ total clusterings for both LSA and ICA with 32 different pairs of clustering solutions to compare.

Evaluation of any clustering output is a difficult task with no widely accepted standard. In many cases, clustering output is evaluated against internal distance- or variance-based criteria such as the *scattering* and *separation* of clusters (He et al., 2002). These measurements do not truly reflect the quality of the output when there is an explicit goal of the clustering such as reflecting a topical organization: they do not offer any interpretation of the clustering output with regard to the stated goal. External methods, on the other hand, offer a direct comparison of the clustering to a pre-existing "gold-standard" classification of the documents, and for this reason can provide a more accurate assessment of the clustering as a tool for topical segmentation of a text corpus. The evaluation metric used in this study will be Normalized Mutual Information (NMI), an information-theoretic criterion which measures the informativeness of the clustering with regard to the pre-existing classification. This metric offers several advantages over other external evaluation metrics such as *purity* and *entropy*: the entire distribution of

documents is evaluated, not just the dominant class; this measure allows the numbers of classes and clusters to vary independently; and it is not biased towards clustering solutions with more clusters (Zhong & Ghosh, 2003). NMI is defined as follows:

$$NMI = \frac{\sum_{i,j} n_{ij} \log(\frac{n \cdot n_{ij}}{n_{i.} n_{.j}})}{\sqrt{(\sum_i n_{i.} \log \frac{n_{i.}}{n})(\sum_j n_{.j} \log \frac{n_{.j}}{n})}},$$

where n is the total number of documents, n_{ij} is number of documents in cluster i and class j, $n_{\bullet j}$ is the number of documents in class j, and $n_{i\bullet}$ is the number of documents in cluster i. NMI is normalized to the range [0,1] and a value of zero indicates a random association between the clustering and gold-standard classification while higher values indicate the clustering solution is more informative of the original classification. A higher NMI value would indicate that the hidden variable discovered by the clustering is more informative of, or more useful in recovering, the original classification. It does not necessarily mean that the clustering mirrors the original classification, although this would certainly result in a high NMI value.

RESULTS & ANALYSIS

This section presents the results of the experiments described above. All the tests in this study were run on a Sun 280R SunFire server with dual 770 Sparc III processors and 4GB of ram. The tests were conducted with custom software utilizing the implementation of the EM algorithm provided by the Weka machine learning toolkit

(Whitten & Frank, 2000). The R toolkit[2] was used for implementations of the ICA and SVD algorithms, using the fastICA[3] and Rserve[4] packages. The complete result set is available in Appendix A.

Several questions will be addressed in the following analysis in addition to the explicit goal of this evaluation. First, a validation of the approach will be presented, demonstrating that the clustering process does indeed discover topical groupings of documents and how documents within an individual clusters can be interpreted as belonging to a single understandable topic. Next, an exploration of the effect of varying the number of dimensions retained in the preprocessing step will be presented. The question of how many dimensions to retain in any dimensionality reduction technique is of keen interest in the tuning of a clustering system. Finally, the evaluation of the difference between Latent Semantic Analysis and Independent Component Analysis as preprocessing techniques will be presented, with an exploration of some of the circumstances when each preprocessing technique may be a better choice for clustering.

Validation of the Clustering Techniques

Several methods will be employed to validate the clustering techniques used in this study. The first of these is an evaluation of the ease of clustering similar groups of documents (the WEB-JV collection) as compared to dissimilar documents (the WEB-JC collection).

[2] http://www.r-project.org
[3] http://cran.r-project.org/src/contrib/Descriptions/fastICA.html
[4] http://stats.math.uni-augsburg.de/Rserve/

Because there is likely to be a higher degree of overlap in the language used in documents about “Java” and “Visual Basic” than in documents about “Java” and “Commercial Banks”, it is anticipated that the clustering of the WEB-JC collection should more effectively separate the two classes of documents than the clustering of the WEB-JV collection. If this is the case, it serves as a validation of the clustering techniques used: if sets of documents with very little topical overlap can be more easily separated than sets of documents with considerable topical overlap, then the clustering techniques are achieving at least some degree of topical grouping.

This theory is borne out in analysis of the results and illustrated in Figure 1. In this Figure, it is clear that the NMI values are considerably higher on average for the WEB-JC collection than the WEB-JV collection, which gives a clear indication that the documents about “Java” and “Commercial Banks” were much more easily separated into those two groups by the clustering.

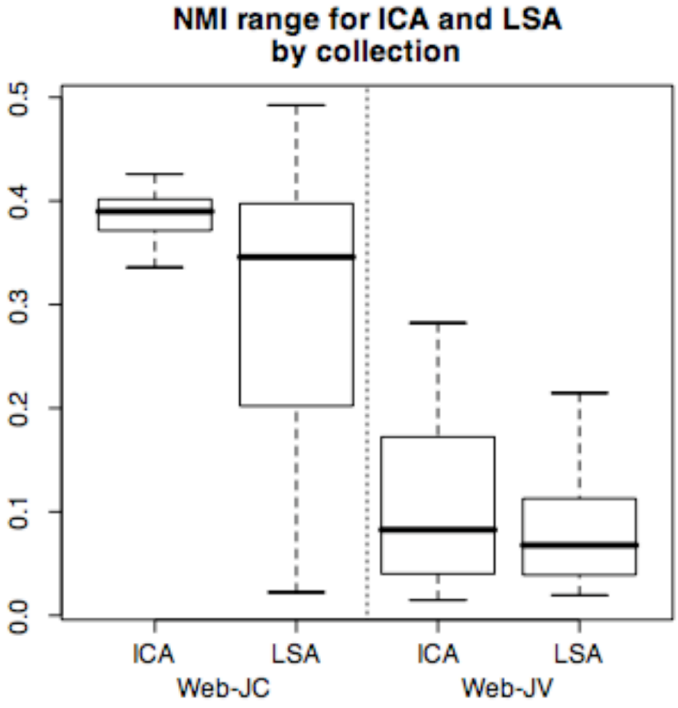

Figure 1 NMI range for ICA and LSA. NMI is significantly higher for WEB-JC collection than the WEB-JV collection, indicating the first is easier to cluster than the second. NMI ranges for ICA and LSA overlap, indicating no clear winner.

As an additional validation measure, looking at confusion matrices and most frequent terms in each cluster can give a good sense of what a cluster is "about". Table 1 shows the confusion matrices for the best clusterings at *k* equal to 5 for both the WEB-JC and WEB-JV collections, and Table 2 shows the ten most frequent terms in each of those clusters. From these Tables, clear associations can be seen between the dominant topic in most clusters and the frequent terms for that cluster. This is an additional confirmation that the clustering process does seem to uncover topical groupings in the document collections.

Confusion Matrix: Web-JC, *k*=5, LSA projection to 10 dimensions					
topic \ *cluster*	*0*	*1*	*2*	*3*	*4*
Java	3	4	520	314	159
Commercial Banks	619	157	6	13	205
Confusion Matrix: Web-JV, *k*=5, ICA projection to 10 dimensions					
topic \ *cluster*	*0*	*1*	*2*	*3*	*4*
Java	96	206	349	306	43
Visual Basic	448	143	3	379	27

Table 1 Confusion matrices for best clustering solutions, *k*=5. This shows a clear tendency for similar documents from the same topic to cluster together.

10 Most Frequent Terms in Each Cluster, stemmed: Web-JC, *k*=5, LSA projection to 10 dimensions				
Cluster 0	*Cluster 1*	*Cluster 2*	*Cluster 3*	*Cluster 4*
bank	bank	java	java	home
service	inform	new	code	bank
account	account	code	program	new
home	provid	program	new	page
inform	secur	creat	creat	contact
rate	appli	user	time	java
person	time	page	tutori	service
save	make	forum	make	click
avail	service	sourc	sourc	access
loan	home	search	user	inform
10 Most Frequent Terms in Each Cluster, stemmed: Web-JV, *k*=5, ICA projection to 10 dimensions				
Cluster 0	*Cluster 1*	*Cluster 2*	*Cluster 3*	*Cluster 4*
basic	new	java	code	code
visual	code	code	home	new
code	program	new	page	program
new	creat	sourc	program	file
program	need	program	new	need
page	applic	user	basic	basic
creat	make	search	site	creat
site	work	page	search	work
link	provid	right	visual	applic
search	time	copyright	download	make

Table 2 Most frequent terms for best clustering solutions, k=5. These top terms can give a sense of what the cluster is "about".

Examination of the frequent terms in each cluster show clear correspondence to the topics in the collection. In the WEB-JC clustering with five clusters using the LSA projection onto ten dimensions, the top half of Table 2, the top terms from the first two clusters clearly show an association with the "Commercial Banks" topic. Terms such as "bank", "account" and "loan" are quite frequent in these clusters, and terms associated with "Java" or programming in general are completely absent. These clusters are comprised of 99% and 97% of documents from the "Commercial Banks" topic. Likewise, the most frequent terms from clusters 2 and 3 show a similarly strong association with the "Java" topic, containing terms such as "java", "code" and "program". The last cluster, cluster 4, shows some interesting characteristics. The document distribution in this cluster is almost equal from each topic, with 44% of the terms from the "Java" topic and 56% from the "Commercial Banks" topic. The frequent terms also seem to be a strange mix, with "home" being the most frequent, and both "java" and "bank" occurring in the ten most frequent terms. An attempt to interpret the "topic" of this cluster is difficult, but several possibilities exist: this cluster could contain documents that didn't fit easily into the first four clusters, thus indicating an "other" topic; or this cluster may contain general administrative- or informational-type pages in both the "Java" and "Commercial Banks" collections, as indicated by the frequency of terms like "home", "click", "information" and "access" as well as the inclusion of both "java" and "bank". A more accurate interpretation of the specific topic of this cluster, if possible, would require a much deeper look into the specific documents within that cluster and possibly other methods of ranking terms highly associated with the cluster. But, although this cluster does not

strongly identify with one of the two pre-existing topics, it is entirely possible that this cluster does represent a coherent topic – a topic that is not isolated to a subset of the "Java" or "Commercial Banks" documents in the collection.

Clustering Performance and Dimensions Retained

Another question that can be evaluated is the effect on clustering performance of the number of dimensions retained, either with ICA or LSA. There are obvious computational advantages to reducing the dimensionality as much as possible, but it is expected that this may be at the expense of the quality of the clustering solution. The optimal dimensionality for document retrieval using LSA is still an open question, although a considerable amount of research has been devoted to this topic (Deerwester et al., 1990; Efron, 2002). This research has generally found that the optimal dimensionality is in the range between 150 and 300 dimensions, and prior to executing these experiments it was expected that a similar range would be useful for clustering. But the experiments conducted in this study show that performance generally decreases as the number of dimensions increase. In fact, of the ten parameterizations that resulted in the highest NMI values, six were with the number of dimensions retained set to ten, the lowest value tested. This comparison can be seen in Figure 2, which shows a clear negative trend in NMI as the number of dimensions increase. The decrease in NMI values indicates that as the number of dimensions increase, the cluster solutions found show a weaker, and more random, association to the original topical classification. Thus, there is a two-fold advantage to reducing the dimensionality: not only do the

computational requirements reduce with the number of dimensions, but the cluster quality actually increases.

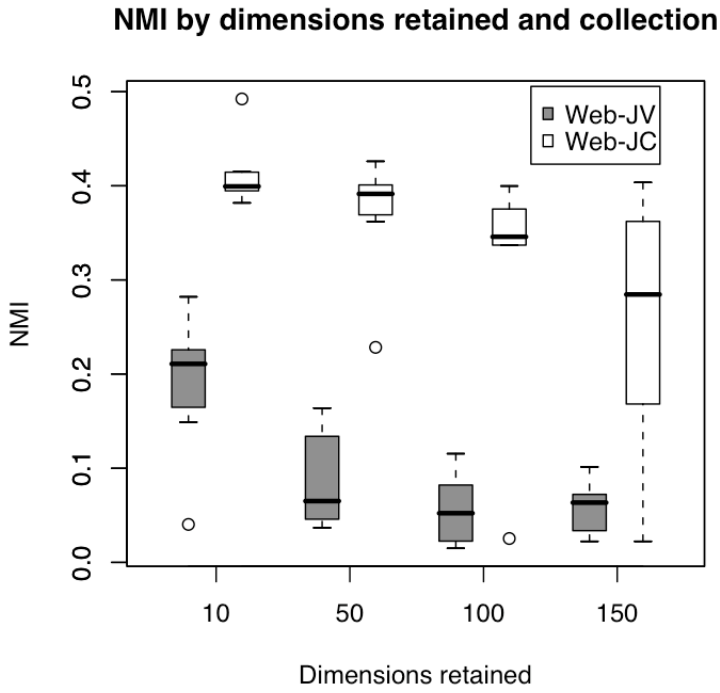

Figure 2 NMI by dimensions retained. A clear negative trend in the NMI values is visible as the number of dimensions increase.

This result, although initially surprising, may make sense when considering the differences between document clustering and retrieval. In a retrieval task, highly nuanced measurements of the degree of similarity between documents and queries must be calculated. In clustering, especially in experiments such as these where the number of clusters is relatively small, a much more general degree of document similarity is calculated. As clustering only requires this higher-level computation, fewer dimensions may retain sufficient data to provide accurate calculations. This analysis may be an

indication that the higher-ranked dimensions in the case of clustering only add noise to the data and therefore degrade performance.

Clustering Performance and Dimensionality Reduction Technique

Finally, comparing the effect of ICA and LSA on clustering shows further interesting and surprising results. It was expected that ICA would clearly outperform LSA, as ICA is specifically identifying dimensions that exhibit a more "clusterable" characteristic. That is, dimensions that are a significant departure from a normal distribution may have a tendency to be the dimensions which more effectively capture the distinctions between clusters. This hypothesis is illustrated in Figure 3, adapted from (Hyvarinen et al., 2001). This Figure is a plot of simulated data from two bivariate normal distributions, and clearly two distinct clusters exist. When reducing the dimensionality of this data from two dimensions to one, the singular value decomposition would project the data to the y-axis, the axis that explains the most variance in the data. ICA, on the other hand, would project the data to the x-axis, that axis that has the least normal distribution. It is clear that, for this data, the ICA projection is the superior choice for clustering purpose: this projection clearly retains the greatest separation between the two distributions whereas the LSA projection eliminates any distinction between the two clusters. This behavior is expected to generalize to higher-dimensional text data and thus the ICA projections are expected to outperform LSA.

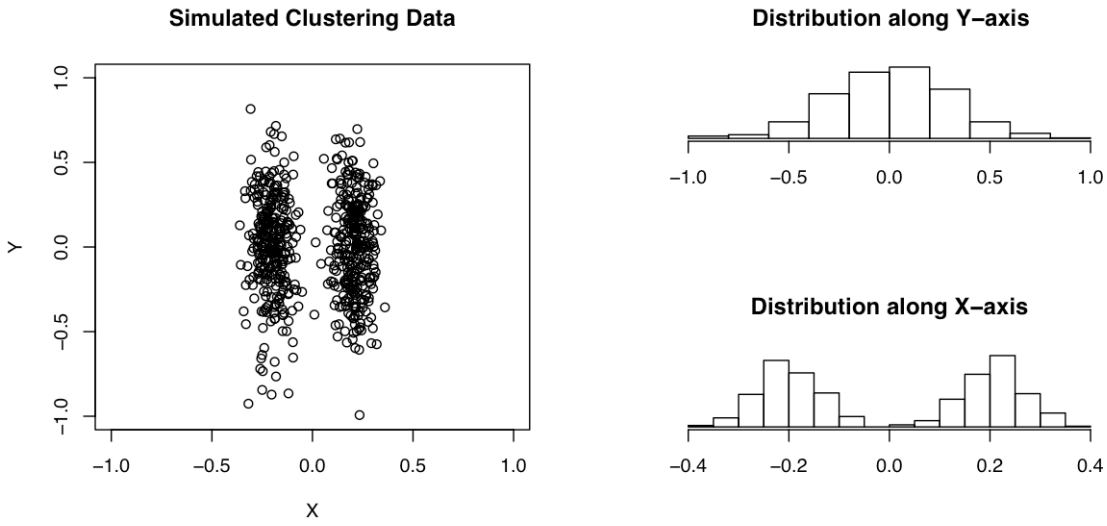

Figure 3 Simulated Clustering Data

Contrary to the indication given by the above admittedly simple example, the test results do not support the hypothesis that ICA generally outperforms LSA for clustering. In fact, the best result from all the test occurred when the LSA projection was used. The range of NMI scores for LSA and ICA can be seen in Figure 1, above. In that Figure it is clear that ICA provides no advantage over LSA in general. Upon closer examination, there are some cases in which ICA does seem to clearly outperform LSA. The following Figures, 4-7 below, show that LSA (the solid lines) and ICA (the dashed lines) perform at approximately the same level when the number of dimensions is low, but the performance of the LSA projection decreases steadily as the number of dimensions increases.

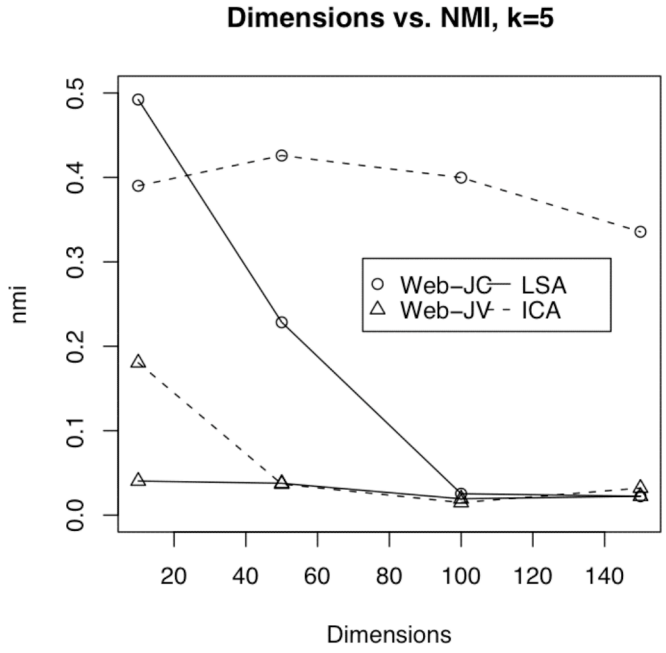

Figure 4 Dimensions vs. NMI, k=5. Higher NMI indicates a better clustering.

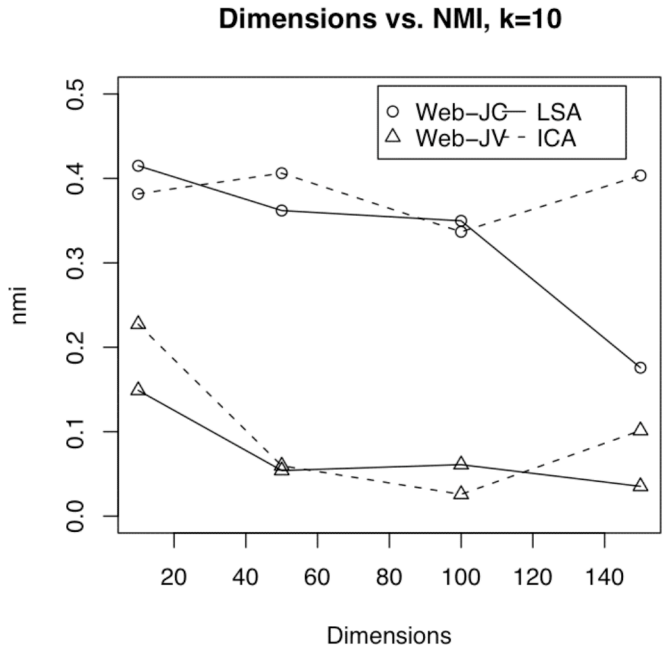

Figure 5 Dimensions vs. NMI, k=10. . Higher NMI indicates a better clustering.

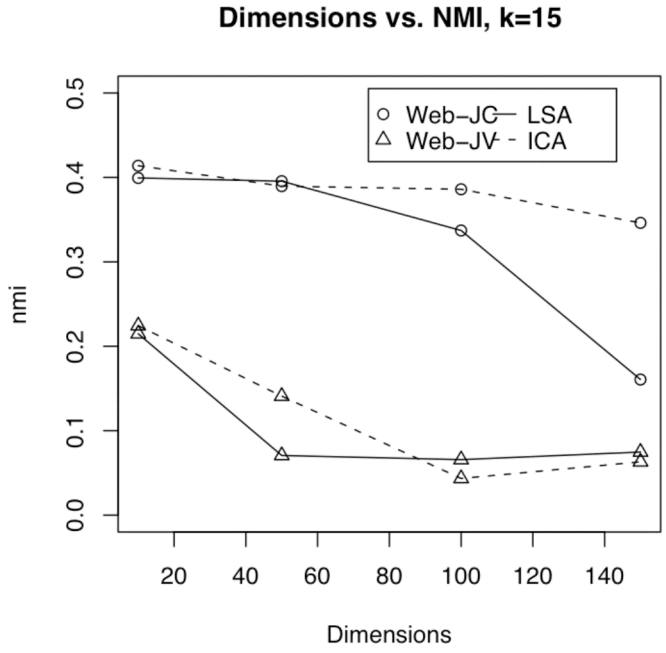

Figure 6 Dimensions vs. NMI, k=15. Higher NMI indicates a better clustering.

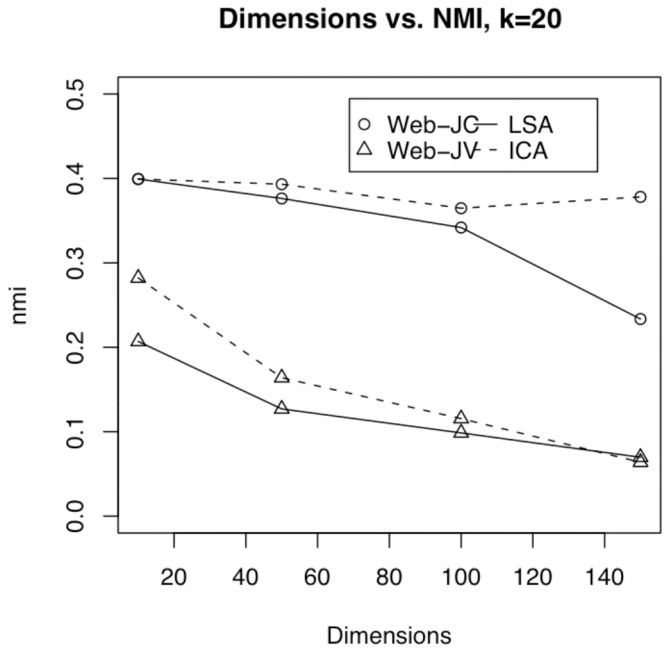

Figure 7 Dimensions vs. NMI, k=20. Higher NMI indicates a better clustering.

These Figures not only show the neck-and-neck performance of ICA and LSA at lower dimensions, but also illustrate the degradation in performance of both ICA and LSA as the number of dimensions increase. In some cases, ICA does not appear to suffer from that problem to the same degree as LSA. In particular, when using the WEB-JC collection, ICA performance appears to be relatively stable as the number of dimensions increase. As stated above, a low NMI value indicates the clustering solution and the original classification have only a random association, whereas a higher value indicates the clustering and the classification are mutually informative. It is clear that LSA outperforms ICA in many cases, and no general statement of the superiority of ICA over LSA can be made.

It is important to note that the performance analysis shown in the above Figures cannot be considered conclusive. Although in some cases there appear to be clear performance trends, these results are based on a very small sample of data. As with many clustering algorithms, the initial random configuration of data used to initialize the EM algorithm can have a strong effect on the final outcome. It is possible to choose a particularly "lucky" (or "unlucky") initial configuration in some of the experiments, and therefore skew the test results dramatically. In order to perform a more rigorous analysis and derive conclusive results, it may be necessary to run a series of tests varying the random configuration to come up with more accurate estimates for each data point. Unfortunately, computational and time constraints prevented this type of analysis from being performed for this study.

CONCLUSION

The experiments conducted for this study attempted to compare the performance of LSA and ICA as dimensionality-reduction techniques for document clustering. Although no definitive answer was reached as to which projection technique performs better, the results do hint at some interesting performance trends. When reduced to 10 dimensions, LSA and ICA appear to perform comparably. At higher dimensions, however, ICA appears to regularly outperform LSA.

In addition to these results, further findings indicate the fewer dimensions retained through either the LSA or ICA projection result in better performance. This finding is surprising: it was anticipated that there would be a trade-off between the amount of information retained and the quality of clusters, but the contrary seems to be the case. Further experimentation is required to elucidate the relationship between the number of dimensions retained, the number of clusters, and other factors affecting the clustering outcome.

BIBLIOGRAPHY

Bingham, E., Kab, A., & Girolami, M. (2003). Topic identification in dynamical text by complexity pursuit. *Neural Process. Lett., 17*(1), 69-83.

Bingham, E., & Mannila, H. (2001). *Random projection in dimensionality reduction: Applications to image and text data.* Paper presented at the ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, California.

Cutting, D., Pedersen, J., Karger, D., & Tukey, J. (1992). *Scatter/gather: A cluster-based approach to browsing large document collections.* Paper presented at the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.

Dasgupta, S. (1999). *Learning mixtures of gaussians.* Paper presented at the 40th Annual Symposium on Foundations of Computer Science.

Dasgupta, S. (2000). *Experiments with random projection.* Paper presented at the 16th Conference on Uncertainty in Artificial Intelligence.

Deerwester, S., Dumais, S., Landauer, T., Furnas, G., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science, 41*(6), 407.

Dhillon, I., & Modha, D. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning, 42*(1/2), 175.

Efron, M. (2002). *Eigenvalue-based estimators for optimal dimensionality reduction in information retrieval.* Unpublished Doctoral Dissertation, University of North Carolina, Chapel Hill.

Efron, M., Elsas, J., Marchionini, G., & Zhang, J. (2004). *Machine learning for information architecture in a large governmental website.* Paper presented at the 4th ACM/IEEE-CS joint conference on Digital libraries, Tuscon, AZ, USA.

Ghosh, J., & Ye, N. (2003). Scalable clustering methods for data mining. In *Handbook of data mining* (pp. 277).

Hand, D., Mannila, H., & Smyth, P. (2001). *Principles of data mining (adaptive computation and machine learning)*.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of statistical learning*.

He, J., Tan, A.-H., & Tan, C.-L. (2002). *Art-c: A neural architecture for self-organization under constraints.* Paper presented at the 2002 International Joint Conference on Neural Networks, Hawaii, USA.

Hearst, M. A., & Pedersen, J. O. (1996). *Reexamining the cluster hypothesis: Scatter/gather on retrieval results.* Paper presented at the ACM International Conference on Research and Development in Information Retrieval.

Hofmann, T. (1999). *Probabilistic latent semantic indexing.* Paper presented at the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.

Honkela, T., & Hyvarinen, A. (2003). *Linguistic feature extraction using independent component analysis.* Paper presented at the International Joint Conference on Neural Networks, Budapest, Hungary.

Hyvarinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis*.

Isbell, C. L., & Viola, P. (1999). *Restructuring sparse high dimensional data for effective retrieval.* Paper presented at the 1998 conference on Advances in neural information processing systems II.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys, 31*(3), 323.

Kolenda, T., & Hansen, L. (1999). Independent components in text.

Leuski, A. (2001). *Evaluating document clustering for interactive information retrieval.* Paper presented at the Tenth international conference on Information and knowledge management.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130-137.

Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval*. USA: McGraw-Hill.

Schutze, H., & Silverstein, H. (1997). *Projections for efficient document clustering.* Paper presented at the 20th annual international ACM SIGIR conference on Research and development in information retrieval, Philadelphia, Pennsylvania.

Sinka, M. P., & Corne, D. W. (2002). *A large benchmark dataset for web document clustering.* Paper presented at the 2nd Hybrid Intelligence Conference, Santiago, Chile.

Steinback, M., Karypis, G., & Kumar, V. (2000). *A comparison of document clustering techniques.* Paper presented at the KDD workshop on Text Mining.

Strang, G. (1993). *Introduction to linear algebra*. Wellesley, MA: Wellesley-Cambridge Press.

van Risjbergen, C. J. (1979). *Information retrieval*: Butterworth.

Whitten, I. H., & Frank, E. (2000). *Data mining*. San Diego, CA: Morgan Kaufmann.

Zamir, O., Etzioni, O., Madani, O., & Karp, R. (1997). *Fast and intuitive clustering of web documents.* Paper presented at the Knowledge Discovery and Data Mining.

Zhao, Y., & Karypis, G. (2001). *Criterion functions for document clustering: Experiments and analysis* (Technical Report No. TR #01--40). Minneapolis, MN: Department of Computer Science, University of Minnesota.

Zhong, S., & Ghosh, J. (2003). A unified framework for model-based clustering. *J. Mach. Learn. Res., 4*, 1037.

Appendix A: Complete Test Results

NMI values for Web-JC collection

		Clusters (k)			
Projection	Dimensions	5	10	15	20
LSA	10	**0.492**	0.415	0.399	0.399
	50	0.228	0.362	0.396	0.376
	100	0.025	0.349	0.337	0.341
	150	0.022	0.176	0.161	0.233
ICA	10	0.390	0.381	0.413	0.399
	50	0.425	0.406	0.390	0.393
	100	0.400	0.337	0.386	0.365
	150	0.336	0.404	0.346	0.378

NMI values for Web-JV collection

		Clusters (k)			
Projection	Dimensions	5	10	15	20
LSA	10	0.040	0.149	0.215	0.207
	50	0.038	0.054	0.071	0.127
	100	0.019	0.061	0.066	0.099
	150	0.022	0.035	0.075	0.070
ICA	10	0.180	0.227	0.224	**0.282**
	50	0.037	0.060	0.140	0.164
	100	0.015	0.026	0.043	0.115
	150	0.032	0.101	0.063	0.064