Representing Documents Using an Explicit Model of Their Similarities

Brian T. Bartell  Garrison W. Cottrell  Richard K. Belew

Department of Computer Science & Engineering
University of California, San Diego
La Jolla, California 92093-0114

Abstract

A method is proposed for creating vector space representations of documents based on modeling target inter-document similarity values. The target similarity values are assumed to capture semantic relationships, or associations, between the documents. The vector representations are chosen so that the inner product similarities between document vector pairs closely match their target inter-document similarities. The method is closely related to the Latent Semantic Indexing approach; in fact, they are equivalent when the target similarities are derived directly from document similarities based on term co-occurrence. However, our method allows for external sources of inter-document semantic constraints to be used in the indexing, though at greater computational expense. The method is applied to three standard text databases from the information retrieval literature. On the CISI database of information science abstracts, performance (measured by precision averaged over a range of recall levels) improves by 28% compared to a weighted term-vector approach, and improves 16% compared to Latent Semantic Indexing. Similar improvement is obtained on the Cranfield database, but no improve-
ment is obtained for the artificial MED database of medical abstracts. The generally favorable performance suggests interesting potential for methods which explicitly modify the retrieval system to meet inter-document semantic constraints.

1 Introduction

A well known and critical problem for document retrieval systems is the failure of individual keywords to identify the conceptual content of documents. It is difficult to determine whether a document is relevant to a query, or whether two documents are conceptually related, simply by examining the terms they share. In this paper we present a method for representing documents which is based explicitly on the semantic relationships between documents in addition to the individual term usage in documents. This allows documents to be retrieved because they are semantically related, rather than simply related by term co-occurrence. We demonstrate experimentally that this can improve document retrieval performance relative to current popular retrieval algorithms.

The problems associated with using document terms as the indices of retrieval are partly the result of the ambiguity inherent in the terms used in natural language text. For example, Furnas, et. al., (1987) have observed that individual keywords are not adequate discriminators of semantic content. Rather, the indexing relationship between word and document content is many-to-many: A number of concepts can be indexed by a single term \textit{(polysemy)}, and a number of terms can index a single concept \textit{(synonymy)}. When retrieval is based solely on the matching of terms between the query and documents, performance suffers. Some relevant documents are missed because they are not indexed by the keywords used in the query, but by synonyms. Conversely, some irrelevant documents are retrieved, because they are indexed by unintended senses of the keywords in the query.
There is currently a great deal of interest in solving this problem of term ambiguity, and a wide variety of approaches have been proposed. Approaches include building thesauri (either automatically or manually) and modifying either document or queries with semantically similar terms (Salton & Lesk, 1971) (Sparck Jones, 1971) (Wang, et. al., 1985) (Nelson, 1993), automatic term disambiguation into term senses (Voorhees, 1993) (McDonald, et. al., 1990), approaches in which document representations are augmented with explicit associations (Belew, 1986, 1989) (Kwok, 1991), and approaches using relevance feedback information to better identify the interests of the user (Harman, 1992) (Salton & Buckley, 1990). All of these methods avoid the simple matching of corresponding keywords in documents to determine relatedness, and attempt some additional semantic extraction or augmentation (e.g., via term expansion) of the text.

A theme common to many approaches to the ambiguity problem is to posit a multidimensional semantic space in which documents are represented. Like the familiar and widely used term-vector representation (Salton, 1983), documents and queries are represented in the space by vectors, and vector similarity is used to determine the semantic relatedness between text objects. When two document vectors are similar, it is interpreted as evidence that the documents are semantically related; when two documents are dissimilar, they are not likely related. In the most typical application of Salton’s approach, there is a dimension in the vector space for each term in the corpus. A document’s vector has non-zero elements only when the corresponding term occurs in the document. Under most typical vector similarity measures (such as inner product or cosine), two vectors are similar only if they share non-zero elements; therefore, only documents which share terms are considered semantically related. Thus, term co-occurrence wholly constrains how the system is able to estimate the relatedness between texts. This is a limitation of the typical application of
Salton’s method, because of the term ambiguity problems already discussed.

A number of enhanced methods have been proposed which have the potential to overcome this limitation. The feature that unifies these methods is that they each represent documents in a multidimensional space such that two document vectors may be similar even if they share no terms. We call this multidimensional space a “semantic space” because it potentially can predict semantic associations far richer than are implied by simple term co-occurrence. Approaches include Latent Semantic Indexing (Deerwester, et. al., 1990), Gallant’s context vector method (Gallant, 1991), Bookstein’s addition of pseudo-terms to model co-relevance structure (Bookstein, 1986), Brauen’s adaptive document vectors (Brauen, 1971), Yang and Chute’s canonical concept mapping (Yang & Chute, 1992, 1993), and Borko and Bernick’s automatic document classification method (Borko & Bernick, 1963).

1.1 Metric Similarity Modeling

The approach we propose, Metric Similarity Modeling (MSM), also uses a multidimensional semantic space in which to represent documents. Semantically related documents are represented by similar vectors, and related documents may be similar even if they share no terms. The difference between our approach and those identified above is the method by which the multidimensional semantic representations are generated. The method has two critical features:

- semantic relatedness between documents, derived from sources in addition to term co-occurrence, is explicitly modeled in the technique by the similarities between vectors in the semantic space, and
• the result of the method is a function which can represent an arbitrary document or query in the semantic space, even if it was not used during the original application of the method.

Whereas numerous techniques modify document representations and implicitly alter the inter-document similarities, few change documents to meet specific constraints derived from the available semantic relatedness information. MSM allows semantic associations to directly determine the interpretation of terms and the representations of text in the multi-dimensional space. In addition, many techniques only derive new representations for the set of documents available at the time the method is applied; the semantic map resulting from MSM can be used subsequently to represent novel documents or queries in the same semantic space.

1.2 Perceived Semantic Similarity in IR

The relative frequency of word occurrences remains the primary source of information concerning the semantic content of texts in an IR corpus, but there are many other sources of data as well. Co-citation information provides one well-investigated example: Two documents can be considered semantically similar to the extent that they tend to cite similar literatures (Everett & Pecotich, 1991) (Garfield, 1979) (Bookstein, 1986). Document classification information, for example the Library of Congress’ subject headings, can also be exploited.

The ongoing stream of relevance assessments coming out of each and every user’s interaction with an IR system promises to provide one the richest sources of data, beyond simple word frequency statistics, concerning documents’ semantic content. The set of documents identified as relevant to the query and therefore “co-relevant” to one another, suggests that
at least one person at one point in time finds them similar relative to the entire set of retrieved documents.

People's perceptions of similarity have been extensively studied within the psychological literature (for example, Rosch, 1973; Tversky, 1977; and Rips, 1989), and much of this background can be brought to bear on the problem of interpreting and exploiting relevance feedback data. The research reported in this paper is one attempt to make explicit this connection to the psychometrics literature, specifically through the application of psychometric Multidimensional Scaling (MDS) methods (discussed in the next section) to the problem of document indexing.

We proceed with an overview and theoretical development of the Metric Similarity Modeling technique. In a comparison of MSM to related techniques, we demonstrate that Latent Semantic Indexing is an important and optimal special case in the MSM framework. We then present the algorithm and discuss its implementation, followed by an application of the method to a set of text collections that are standard in the information retrieval literature. We conclude with a discussion of the strengths and weaknesses of the proposed method.

2 Derivation of Method

The MSM approach places primary emphasis on inter-document similarities. The actual document representations are subservient: They are adjusted so that the similarities between document representations can take on appropriate desired values. This kind of approach is typical in the broader field of Multidimensional Scaling (MDS) (Green, et. al., 1989) (Borg & Lingoes, 1987): In MDS, objects are represented as points in a multidimensional space; points are chosen so that the inter-point similarities meet a set of externally
imposed constraints on the similarities. Our MSM approach is derived directly from well known techniques in MDS. In the following sections we provide an overview of MSM using concepts from Multidimensional Scaling theory, and formally present the MSM problem and solution.

2.1 Overview

Metric Similarity Modeling addresses the well-known problem that the co-occurrence of terms between documents is only a limited estimate of the semantic relatedness between documents. The problem is addressed in MSM by explicitly modeling additional inter-document semantic relatedness information which may be available. The semantic information is modeled by MSM by the similarities between vectors in a multidimensional “semantic” space. Vectors which are similar correspond to semantically related documents; vectors which are not similar are not semantically related. In addition, MSM generates a representation function which can be used to represent either the original documents or an arbitrary document or query as a vector in the semantic space. The representation function takes as input a document or query represented as a vector of terms, and produces an appropriate semantic vector.

A schematic of this process is provided in Figure 1. Here, documents in the collection are initially represented as vectors in Salton’s traditional multidimensional term space model (Salton, 1983). Externally available semantic information, in the form of an inter-document association matrix, is used to derive a function which maps documents from the term space to an alternative representation space. The new space, a “semantic space”, represents documents so that the inter-document associations are more accurately predicted by the inter-vector similarities.
Figure 1: MSM generates a representation function which maps term-space document vectors into vectors in a multi-dimensional semantic space. The representation function is chosen so that the inner product similarities between the semantic document representations closely match a set of target similarity constraints.

The main value of the method is that related documents will be represented similarly and will therefore tend to be retrieved together. Documents relevant to a query will be retrieved even when they share no terms with the query, since they are represented similarly to other relevant documents. The result is a significant improvement in precision, especially at high levels of recall.

2.1.1 Similarity Constraints

To perform the representation task, MSM requires two kinds of information:

- a term-vector encoding of the documents to be represented, and
- a separate indication of the semantic relatedness between the documents.
We call the semantic relatedness information similarity constraints because they are used to constrain the representation of documents. MSM explicitly satisfies the similarity constraints by deriving document representations having inter-vector similarities which are as close as possible to the similarity constraints.

The similarity constraints used by the Metric Similarity Modeling algorithm must be both metric and exhaustive. A metric similarity constraint for a pair of documents provides a specific target value, a real number, which should be matched by the inner product similarity between the two documents in the new representation. For example, let the metric similarity constraint for documents 2 and 4 from some collection be 0.3. Then the similarity between the vectors for documents 2 and 4 in the semantic space is constrained to be 0.3. Note that there are many ways in which the vectors for documents 2 and 4 could be positioned in the semantic space and still satisfy this constraint; one possible configuration is illustrated in Figure 1.

The similarity constraints must be exhaustive in that a target similarity value is required for all pairs of documents to be automatically indexed by the method. Once the method is applied and a representation function is derived, arbitrary documents or queries can be represented in the semantic space. No target similarity constraints are needed for these novel cases. However, when new documents with additional semantic constraints are added, MSM is re-applied to this new collection.

Similarity constraints are a non-standard component of typical document indexing strategies, making MSM only applicable where such information is available. However, where these constraints are available, MSM provides a technique for providing semantic document representations which satisfy them. We discuss in section 5 an approach to automatically generate the target similarities from document relevance data. Other sources
of target similarity data may also be available or suitably derived.

2.2 Problem Statement and Analytic Solution

In this section, we formally define the representation problem and give the solution provided by the MSM procedure.

The problem is to find a linear mapping which optimally re-represents term-vector documents as vectors in a new vector space (a semantic space), such that inner product similarities in the semantic space most closely match target inter-document similarities. Formally, let $X$ be a real matrix with $t$ rows and $d$ columns (denoted $t \times d$), where $t$ is the number of terms in the term-vector representation and $d$ is the number of documents in the corpus. The $i$'th column of $X$ corresponds to the $i$'th document in the collection, and the $j$'th row corresponds to the weight of the $j$'th term in that document. Thus, $X$ contains the complete term-vector representation of all documents in the collection. Let $S$ ($d \times d$) be a square, symmetric, positive semi-definite matrix containing the target inter-document similarities. The $i$, $j$'th element of $S$ is the target similarity of the $i$'th and $j$'th documents. Let $k$ be the dimensionality of the target semantic space. The representation function will map documents in the $t$ dimensional term space to new representations in a $k$ dimensional semantic space. Let $W^{(k)}$ ($k \times t$) be the real matrix implementing this linear representation function. $W^{(k)}$ will be the result of the MSM procedure. The superscript $(k)$ on $W^{(k)}$ emphasizes that $k$ is a free parameter in the method. For the results reported in this paper, $k$ is fixed at 100 and yields consistently satisfactory results. 100 is also the dimensionality used by Deerwester, et. al., when applying the related Latent Semantic Indexing (Deerwester, et. al., 1990) technique to a number of databases.

\footnote{A symmetric positive semi-definite matrix $M$ is one for which a real matrix $B$ exists such that $M = B^T B$.}
The product \( W^{(k)}X \) is a \( k \times d \) matrix of documents represented in the semantic space. The goal is for the inner products between these representations to match the target constraints. Let \( \hat{S} \) be the inner product similarities between the documents in the semantic space, i.e., \( \hat{S} = (W^{(k)}X)^T(W^{(k)}X) \), where \( T \) is the matrix transpose operator. The problem is to find the \( W^{(k)} \) which minimizes the error criterion:

\[
E = \sum_{i=1}^{d} \sum_{j=1}^{d} (\hat{S}_{i,j} - S_{i,j})^2
\]

This measure of error is simply a sum of squared differences between the corresponding elements of \( S \), the target similarity matrix, and \( \hat{S} \), the similarities between the documents re-represented in the semantic space.

We demonstrate in the Appendix (with further details in Bartell, et al., 1992) that a critical point of equation (1) is of the form:

\[
W^{(k)} = M^T_{k} C X^+
\]

Here, \( C \) is found using the Singular Value Decomposition\(^2\) (SVD) of \( S \) into \( S = C^T C \). \( X^+ \) is

\(^2\)Singular Value Decomposition, or SVD, (Strang, 1988) is a numerical analysis tool which has great utility in the solution of a number of MDS problems. For an arbitrary real matrix \( X \), the SVD uniquely (up to certain trivial re-arrangements of columns and sub-space rotations) decomposes the matrix into the product of three matrices:

\[
X = U L A^T
\]

The three matrices have a special restricted form. \( U \) and \( A \) are both column orthonormal; that is, the columns are orthogonal (i.e. the inner product of two different columns is 0) and are unit length. \( L \) is a diagonal matrix of singular values: all off-diagonal elements are zero; all diagonal elements (the singular values) are non-negative real values, typically ordered in decreasing value. When \( X \) has \( t \) rows and \( d \) columns (denoted \( t \times d \)) and is rank \( r \), then we allow \( U \) to be \( t \times r \), \( A \) to be \( d \times r \), and \( L \) to be \( r \times r \) with no zero singular values.

The SVD has many important properties. Foremost for the purposes of this exposition is that it provides the best lower rank approximation of a matrix \( X \) in terms of the Euclidean matrix norm (or Frobenius norm, calculated by taking the square root of the sum of all squared entries of a matrix) (Stewart, 1973). More precisely, let \( U_k \) be the \( t \times k \) (\( k \leq r \)) matrix found by removing \( r - k \) columns from \( U \). The \( k \) columns remaining in \( U_k \) correspond to the largest singular values in \( L \) (similar versions of \( L_k \) and \( A_k \) can be defined). Then \( \hat{X} = U_k L_k A_k^T \) minimizes the sum of squared differences between corresponding elements of matrices \( X \) and \( \hat{X} \).
the pseudo-inverse of $X$ (Strang, 1988). The pseudo-inverse can also be calculated using the SVD: Let the SVD of $X$ be $X = ULA^T$; then $X^+ = A L^{-1} U^T$, where $L^{-1}$ denotes a diagonal matrix having the reciprocals of the singular values in $L$ along the diagonal. $M$ is provided by the Singular Value Decomposition of the product $CAA^T$ into $CAA^T = MΣN^T$, where again $A$ is given by the SVD of $X$. $M_k$ is derived from $M$ by retaining the $k$ columns of $M$ corresponding to the largest singular values in $Σ$ and discarding the rest. Section 4 provides an algorithmic interpretation of this result.

Though this solution may seem cryptic, we are essentially solving the familiar least squares problem

$$W^{(k)}X = C_k$$

and solving for $W^{(k)}$. Here, $C_k$ denotes the $(k \times d)$ matrix found by keeping the $k$ rows of $C$ corresponding to the largest eigenvalues of $S$ and removing the rest. It is well known in the data analysis literature that $C_k$ provides the optimal representation of documents such that the inner-products between the objects in $C_k$ best match the target similarities given by $S$. Thus, the least squares problem can be interpreted as finding the best linear map from the term-vector representation to the optimal $k$ dimensional representation. Unfortunately, in detail the solution is somewhat more complex than this; these details are provided in the following section.

### 2.2.1 Details of the Solution

It would be fortunate if the least squares solution to

$$W^{(k)}X = C_k$$


would give the optimal solution to our metric similarity modeling problem: Since $C_k$ is the optimal document representation, the least squares solution would give the best linear map to that optimal representation. However, like all linear least squares problems, there may be no $W^{(k)}$ such that the equality can be satisfied. For example, from linear algebra we know that when the row space of $C_k$ is not entirely a subspace of the row space of $X$, we cannot solve the equality.

Complicating this problem is the fact that some of the rows of $C$ which were discarded when constructing $C_k$ may in fact intersect the row space of $X$. In this case, the singular vectors which correspond to the smaller singular values have information which correlates with the target similarity information. As an example, consider an extreme case: The row spaces of $C_k$ and $X$ are orthogonal, but the row spaces of $C_{r-k}$ and $X$ are identical (here, $C_{r-k}$ denotes the $(r - k) \times d$ matrix constructed from the $r - k$ smallest singular vectors and values of $C$, with $r$ the rank of $C$). In this pathological case, the most significant dimensions of term co-occurrence do not coincide with the optimal semantic representation $C$, so no linear mapping between the two can be found. That is, the least squares solution to $W^{(k)}X = C_k$ is $W^{(k)} = 0$. However, there is an exact solution to $W^{(k)}X = C_{r-k}$ which provides a better solution to the error we are minimizing in (1), and which therefore provides a better approximation to the optimal semantic representation.

The solution given in equation (2) reflects this complication of the simple least squares approach. The matrix $C$ is projected onto the row space of $X$ and the spanning vectors (i.e., the singular vectors) of this matrix are used in place of $C_k$ in the least squares problem. In the special case when $C_k$ is a subset of the row space of $X$, the projection does nothing and the solution is the simple least squares. In the general case, we solve the modified least
squares problem

\[ W^{(k)}X = M_k^T C \]  

(6)

where \( M_k^T C \) keeps the most significant dimensions of the optimal representation \( C \) once projected onto the row space of \( X \).

### 3 Latent Semantic Indexing and MSM

Latent Semantic Indexing (LSI) is a document indexing technique closely related to the MSM method advocated here. Latent Semantic Indexing begins with a term-vector representation of documents, and attempts to improve retrieval performance by re-representing both documents and terms in a semantic space analogous to the semantic space used by MSM. According to Deerwester, et. al., (1990), this new document representation has the advantages that

- the dimensions in the space are uncorrelated (i.e., they are orthogonal),

- the representations are less noisy than the original term representations, and

- the representations incorporate higher-order (latent) association structure among terms and documents.

These properties are a result of the technique used to re-represent the documents in the lower dimensional space. The technique used is Singular Value Decomposition, applied to the same \((t \times d)\) document matrix \( X \) used in the previous section. Latent Semantic Indexing employs the SVD of \( X \) to derive the semantic representations of documents. If the SVD of \( X \) is \( X = ULA^T \), then row \( i \) of \( A_kL_k \) gives the representation of the \( i \)’th document in the \( k \) dimensional semantic space. These re-representations of the documents are used in place of
the original \( t \) dimensional term-vector representations when making similarity judgements.

The matrix \( U_k^T \) plays a role in Latent Semantic Indexing analogous to the linear mapping function in MSM. Novel documents or queries can be represented in the semantic space using \( U_k^T \), assuming they are available as vectors in the \( t \) dimensional term space. Let a query be encoded as a column vector \( q \) in \( \mathbb{R}^t \). Then the \( k \) dimensional semantic representation of the query is given by \( U_k^T q \). The similarity of each document in the corpus to a query \( q \) is then found by measuring the similarity between rows of \( A_k L_k \) and the vector \( U_k^T q \). The similarity can be measured using, for example, the cosine measure, as was done in the original work (Deerwester, et. al., 1990).

### 3.1 LSI is an Optimal Case of MSM

We consider now a restricted case of the Metric Similarity Modeling problem. In this case, the metric target similarities are derived directly from the documents themselves. We show here that the optimal MSM solution for this special case is identical to the document indexing solution provided by Latent Semantic Indexing. Thus, Latent Semantic Indexing is generating a semantic space which is modeling the similarity structure of the full term space. Implications of this result for both Latent Semantic Indexing and MSM are discussed in the following section.

We now detail the equivalence between LSI and a special case of MSM. Let \( X \) again be the \((t \times d)\) matrix of \( d \) documents represented using \( t \) terms over \( \mathbb{R}^t \), with Singular Value Decomposition \( X = U L A^T \). The similarity \( s_{i,j} \) between a pair of documents \( i \) and \( j \) (represented by columns \( x_i \) and \( x_j \) of \( X \)) using the inner product similarity measure is then \( s_{i,j} = x_i^T x_j = \sum_{k=1}^t x_{i,k} x_{j,k} \). The matrix \( S \) \((d \times d)\) of all pairwise similarities is given by \( S = X^T X \).
We now apply MSM for the documents $X$ and the target similarities given by $S$. The similarity matrix $S$ satisfies the condition that it be positive semi-definite by definition, since $S = X^T X$. Applying MSM we find that the optimal linear weighting function is given by $W^{(k)} = M_k^T C X^+$, where the matrices $M_k^T$ and $C$ are as defined in section 2.

Because of the special relationship between the target similarities $S$ and the documents $X$, the matrices $M$ and $C$ can be easily solved for in terms of the decomposition matrices of $X$. In particular, $C = X$, and $M = U$. Using the definition of the pseudo-inverse of $X$, $X^+ = A L^{-1} U^T$, and properties of orthonormal matrices, we can then solve for $W^{(k)}$ directly:

$$W^{(k)} = M_k^T C X^+$$
$$= U_k^T X X^+$$
$$= L_k A_k^T X^+$$
$$= U_k^T$$

Thus, MSM yields a linear function $W^{(k)} = U_k^T$ as a solution to this special problem in which the target similarities are given by the inner product similarities of the documents themselves. This function is identical to the solution given by Latent Semantic Indexing applied to the documents.

This result should not at all be surprising. It is well known in the data analysis literature that the $k$ dimensional representation of data which best matches a target matrix of inner product similarities is given by the SVD of the similarity matrix. In this case, with the targets trivially derived from the documents themselves, there is a particularly simple linear relationship between the documents and this optimal representation.
3.2 Implications of the Optimality of LSI

The optimality of Latent Semantic Indexing with respect to this special case of MSM has important implications for both LSI and MSM. For MSM, it provides a practical foundation for application of the technique. That is, if the similarities between the original term-vector documents are used as targets, the method will yield the same favorable results reported for applications of LSI. In addition, this foundation can be used as a starting point for the method when determining the target similarities for a collection of documents: The document vector similarities can serve as default target similarities, augmented by additional available semantic information. Section 5.3 provides a detailed example of this approach, in which co-relevance assessments are used to augment a default target similarity matrix.

The observation that LSI is optimally solving the metric MDS problem complements the analysis of other researchers. Previous work (Deerwester, et al., 1990) has illustrated beneficial properties of the technique (such as reduction of noise, orthogonalization of the vector space, and incorporation of associational relationships in the representation), and the equivalence with Metric Similarity Modeling does not detract from these results. Rather, the current analysis adds new terminology and an alternative perspective to the discussion.

One major insight, given by this analysis, is the importance of the inner product similarities in the original term space. When these similarities are a good, though perhaps noisy, estimate of the relatedness between documents, the LSI reduction should yield good results.

The analysis further suggests that certain term weightings in the original term space should improve the performance of LSI. It is well known that alternative weightings of the terms in a term space representation can improve retrieval performance (Harman, 1986) (Salton & Buckley, 1988). The authors of LSI permit such alternative term weightings
(Deerwester, et. al., 1990). Essentially, these alternative weightings yield a new document matrix $X$ with its own similarity structure $S = X^T X$ based on inner products. When $S$ is a better measure of relatedness than $S$, our analysis predicts that LSI will likely perform better when operating on $X$ than on $X$. Dumais’ empirical study of LSI using various term weightings seems to agree with this suggestion (Dumais, 1990): Term weightings which tend to improve inner product retrieval in the original term space also tend to improve retrieval performance in the semantic space.

In past applications of Latent Semantic Indexing, similarity in the semantic space has been measured by the cosine measure rather than by inner product. Our result does not appear to generalize when cosine is used: We have not shown that cosines in the original term space are preserved in terms of cosines in the semantic space. We certainly have not shown that cosine is an inappropriate measure. Rather, the analysis demonstrates that inner product is a natural measure, when it is desirable to preserve the inner products from the original space. Cosine may also be a useful and appropriate measure because the document vectors generated by Latent Semantic Indexing will not likely be unit length, even if the original document vectors are normalized. The cosine measure will ensure that no document is more likely to be retrieved simply because its semantic representation is longer than others. In addition, it is possible to preserve cosine similarities in the original space in terms of inner products in the semantic space. This is achieved by normalizing the documents first in the term space (resulting in a new document matrix $X$). Since cosines and inner products are identical with unit length vectors, cosines are preserved as inner products by LSI. Finally, our analysis does not preclude the use of other typical similarity measures, such as Pseudo-Cosine or Dice (Jones & Furnas, 1987). Rather, the analysis indicates that these other measures will be applied to a semantic representation which
has much in common with the original space, in terms of their inner product similarity structures.

4 Indexing and Retrieval Algorithm

The solution to the Metric Similarity Modeling problem presented in section 2 is constructive and amenable to implementation. This section details the actual application of MSM to document indexing and retrieval. We assume for now that the target similarities (the inter-document semantic information) are provided; however, in section 5.4 we discuss one method for deriving these values from relevance feedback samples. There are two phases to the application of MSM: Indexing, and retrieval. Indexing is a “training phase” in which document representations are learned from their term space representations and from the target similarities. The primary product of the training phase is a linear map which can represent an arbitrary term-space document or query as a vector in the semantic space. Once trained, the map is used to semantically represent queries, which are then matched against the collection of semantic document vectors.

4.1 Computing Document Indices

The result of the MSM procedure is a linear map, which re-represents objects in the term-vector space as objects in the $k$ dimensional semantic space. The linear map is represented by a matrix $W^{(k)}$, having $k$ rows and $t$ columns, $t$ the number of terms. The matrix product of this weight matrix and a term-space document (represented as a column vector) gives the representation of the document in the $k$ dimensional semantic space. The parameter $k$ is analogous to $k$ in the Deerwester, et. al.,’s (1990) Latent Semantic Indexing work: $k$ is the number of dimensions in the new semantic space. $k$ is always less than or equal to the
rank of the original document space (and is therefore less than or equal to the minimum of $t$ and $d$, the number of documents).

Application of MSM to derive $W^{(k)}$ involves three uses of the Singular Value Decomposition (SVD) technique. As before, we let $X$ ($t \times d$) be a matrix of $d$ documents (the columns of $X$) represented using $t$ terms (the rows of $X$). Furthermore, let $S$ ($d \times d$) be a square, symmetric, positive semi-definite matrix containing the target similarity values (i.e., entry $S_{i,j}$ is the target similarity for documents $i$ and $j$). The three main steps in the MSM procedure are:

1. Compute $\text{SVD}(X) = U L A^T$.

   Note that this is identical to the computation required in LSI. $U$ and $A$ are column orthonormal matrices, and $L$ is a diagonal matrix of singular values.

2. Compute $\text{SVD}(S) = C^T C$.

   Since $S$ is positive semi-definite, $C$ is guaranteed to exist. Recall that $C_k$ provides the optimal $k$ dimensional representation of the data with respect to modeling the target similarities.

3. Compute $\text{SVD}(CAA^T) = M \Sigma N^T$.

   This step determines how the space spanned by the terms of the documents overlaps with the optimal semantic space, $C$.

   Our solution $W^{(k)}$ is calculated directly from the results of the preceding SVD steps:

   $$W^{(k)} = M_k^T C X^+$$  \hspace{1cm} (8)

   Again, $M_k$ is derived from matrix $M$ (in step 3) by retaining the $k$ columns of $M$ corre-
sponding to the largest singular values in $\Sigma$ and removing the other columns. $X^+$ is the pseudo inverse of $X$, i.e., $X^+ = AL^{-1}U^T$. As earlier, $L^{-1}$ is derived from $L$ by taking the reciprocal of all non-zero singular values along the diagonal of $L$.

Given $W^{(k)}$, the new document representations are trivial to compute. They are:

$$X^{(k)} = W^{(k)}X$$  \hspace{1cm} (9)

Recall that the columns of $X$ are the term-vector representations of the documents. Thus, to compute the semantic representations for the document collection, we compute the product of the linear mapping function $W^{(k)}$ and each column of $X$ in turn. The columns of the resultant matrix $X^{(k)}$ are the $k$ dimensional representations of each of the documents.

### 4.2 Comparing Documents and Queries

To estimate the most relevant documents for a particular query, the query is mapped into the same semantic space in which the documents are represented. Then, the query and document vectors are compared using a measure of vector similarity. The most similar documents to the query vector are returned to the user as most likely to be relevant to the query.

It is simple and efficient to map the query into the semantic space. We assume that the query can be represented in the same $t$ dimensional term-vector space as are the original documents. We label this query vector $q$, and let it be a column vector (i.e., a matrix with $t$ rows and a single column). Then the new semantic vector, $q^{(k)}$, is given by the matrix product of the linear mapping function, $W^{(k)}$, and $q$:

$$q^{(k)} = W^{(k)}q$$  \hspace{1cm} (10)
4.3 Computational Cost

There are two costs to using the MSM algorithm: The first cost is paid only once during the indexing (or training) phase; the second is paid for each retrieval using the new representations. The indexing cost is essentially the cost of calculating $W^{(k)}$, which is mostly determined by the cost of computing the Singular Value Decomposition the three times. Algorithms for the SVD are typically order $O(n^3)$, where $n$ is the number of rows or columns of the matrix. In the case of MSM, $n$ is the number of documents ($d$), as we can always compute the SVD of symmetric ($d 	imes d$) matrices. Certain steps in the indexing algorithm also require the matrix multiplication of document matrices; these operations are order $O(d^2t)$ and $O(t^2d)$, where $t$ is the number of terms. For a large number of terms, this may overshadow the SVD cost. The cost at retrieval time is not as burdensome, fortunately.

Computing the semantic representation for the query takes $O(kt_q)$ time, where $t_q$ is the number of terms in the query. Finding the similarity of every document vector to the query vector requires $O(kd)$ time (a computation which can also be done in parallel), and finding the $r$ highest ranked documents takes $O(d \log r)$.

Latent Semantic Indexing has similar indexing and retrieval costs, as it also computes the Singular Value Decomposition of the document matrix. Thus, indexing in LSI is similarly expensive in comparison to fast methods like term weighting (though LSI requires only one SVD whereas we require three). However, unlike in the MSM algorithm, LSI can benefit from certain optimizations due to the limited number of dimensions in the resulting semantic space, and due to the typical sparseness of the document matrix. As Deerwester, et.al., (1990) report, an $O(n^2k^3)$ algorithm can be used, with $k$ the number of dimensions in the semantic space. An alternative is the use of Lanczos algorithms (Cullum & Willoughby, 1985), which make optimized use of the very sparse form of typical document matrices.
5 Experiments

Experiments using three standard text bases from the information retrieval literature are used to validate the Metric Similarity Modeling approach. The experiments test whether retrieval performance can be improved by using representations generated by MSM as compared to two benchmark retrieval methods: A weighted term-vector approach (weighted by the SMART retrieval system), and Latent Semantic Indexing. The three standard collections, the CISI collection of information science abstracts, the Cranfield collection, and the MED collection of medical abstracts, are used to test the methods. They are chosen because they are commonly used to test IR system performance in the literature, standard IR approaches have had mixed success on these collections, and because the size of each collection (approximately 1500 documents) is tractable for the SVD operations. Included with each of the test collections is a set of evaluation queries. The performance of an arbitrary retrieval method can be estimated using these queries, since all documents in the collection have been identified as either relevant or irrelevant to each query.

In our experiments, the queries and accompanying relevance assessments must be used for two distinct purposes. The relevant sets of documents associated with each query are first used as the basis of inter-document similarity information required by the MSM method. The need to use these same queries as the basis of performance evaluation remains, however. This issue is common in machine learning research, where the typical solution is to partition the available data into two sets, an initial “training” set used to construct a representation, and the remaining “test” set used to evaluate it. In the case of the query/relevance sets used in our experiments, the training set is used to provide document similarity constraints and the test set is used to evaluate the resulting MSM representation. Unfortunately, this procedure makes the performance evaluation sensitive to the queries selected for training.
Our experiments therefore iterate this procedure over multiple, random partitions of the query data and then average the results. The resulting procedure, for each of the three test databases, is:

1. Apply the benchmark weighting methods:
   
   (a) Generate weighted term space representations of the documents and queries using SMART.
   
   (b) Generate LSI representations from the SMART weighted vectors following the LSI procedure given in Section 3.

2. Iterate a number of trials:

   (a) Randomly partition the set of all queries into a training set and a test set.
   
   (b) Use the training queries to derive the target similarity constraints for the MSM procedure (we discuss our method below).
   
   (c) Generate MSM representations from the SMART weighted vectors following the MSM procedure.
   
   (d) Measure the performance of MSM, LSI, and SMART using the test queries.

3. Compute MSM, LSI, and SMART performance, averaged over all random query partitions.

All three indexing methods use the same initial document data – the document vectors generated by the SMART retrieval system. This ensures that there is no bias from using different term sets, weighting algorithms, etc., as input to the different methods. In addition, our application of MSM uses queries with relevance-tagged documents to derive the target similarity constraints. The performance of the methods is measured using a set of queries
(the test set) that are not used during indexing. This ensures that there is no bias by evaluating the MSM algorithm on a set of queries that are used in order to determine document representations. Finally, the total query set is randomly partitioned into test set and training set a number of times. This ensures that no bias is introduced by a single poorly randomized partition.

We proceed with a discussion of the major steps in this test procedure. Section 6 provides results of the experiments.

5.1 Benchmark Weighting Methods

The SMART retrieval system is used to generate the weighted term-vector representations. For the MED database, weights are calculated using the TFIDF (Term Frequency–Inverse Document Frequency) method\(^3\). For the CISI and Cranfield databases, the Augmented TFIDF weighting method\(^4\) is used. In all three collections, document and query vectors are normalized to have unit L2 norm (thus, the inner product and cosine similarity measures are identical over the representations). These weighting methods are chosen because they are default settings for the SMART system for these databases, and they result in good performance relative to other possible weightings.

Document representations for the Latent Semantic Indexing method are calculated using the procedure reported in Deerwester, et. al. (1990), except the SMART weights are used in the document matrix rather than raw term frequencies. Experiments using LSI and different term weights indicate that LSI performance can be increased by using weightings which have better performance in the complete term space (Dumais, 1990). Again, theoretical

\(^3\)The TFIDF weight is the term frequency multiplied by the log of the quotient of the number of documents and the collection frequency of the term.

\(^4\)Augmented TFIDF is identical to TFIDF except the term frequency is normalized between 0.5 and 1.0.
justification for this is provided in Section 3. The dimensionality of the semantic space is chosen to be 100 and the similarity metric used is cosine; both of these are used in the original LSI research.

5.2 Partitioning the Query Set

The query set is partitioned into two disjoint sets, the training set and the test set. The training set is used to construct the target similarity constraints used in the MSM procedure. The test set is reserved for later use during evaluation.

The query set is partitioned randomly so that 2/3 of the queries are placed in the training set, and the remaining 1/3 are placed in the test set. The MED collection has 30 sample queries; thus, each partition has 20 training queries and 10 test queries. The CISI collection has 76 sample queries, partitioned into 51 training and 25 test queries. Cranfield has 225 queries, partitioned into 150 training and 75 test queries. Multiple partitions are created for each collection: 8 different partitions are created for the CISI collection, 5 for Cranfield, and 5 for MED. The results reported are the averages over all partitions.

5.3 MSM Weighting

The document representations for the MSM method are calculated using the procedure in section 4. As with Latent Semantic Indexing, the SMART term weights are used to populate the document matrix \( X \). The dimensionality of the MSM semantic space is also 100, like LSI, to avoid bias due to differences in the representational capacity. Also like LSI, the cosine measure is used to determine the similarity between document and query vectors. We have found that the performance of the cosine measure is superior to that of the inner product measure. The use of cosine is appropriate because the document vectors
generated by MSM are not all unit length; cosine normalizes the vectors and makes the prior likelihood of retrieval of each document uniform.

To apply MSM, a matrix of target inter-document similarities, \( S \), is required. This data requirement is additional relative to the requirements of term-vector weighting and LSI. This matrix is what allows MSM to incorporate sources of information other than term co-occurrence to affect the retrieval performance.

There are numerous potential sources of inter-document relatedness information, such as co-citations and hierarchical structures on the collection. This topic is discussed further in section 7. The experiments here use co-relevance information to derive a target inter-document similarity matrix. Two documents are co-relevant when they are both classified as relevant to the same query. Since co-relevance is an indication that two documents are related, at least with respect to a query, we consider this a reasonable source of information to indicate that two documents may be more similar than their base rates of term co-occurrence may indicate.

### 5.4 Construction of Target Similarities

The procedure to calculate a matrix of target similarities involves three steps: generating a baseline similarity matrix directly from term co-occurrences, augmenting the entries in this matrix using co-relevance information, and normalizing the matrix.

The baseline similarity matrix is generated simply by calculating the similarity between all pairs of documents using the inner product measure. This provides a similarity matrix identical to the theoretical target matrix used in the Latent Semantic Indexing. Note that since all documents have been normalized to be unit length, the maximum similarity for any pair of documents is 1.0 and occurs when the documents are identical. The minimum
similarity is 0.0 and occurs if they share no terms. The resultant similarity matrix is symmetric with all diagonal elements (corresponding to the similarity of documents to themselves) equal to 1.0.

Co-relevance data from the training queries is used to augment entries in this baseline matrix. For each pair of documents $i$ and $j$ that are co-relevant to a query, the $i,j$'th entry of the similarity matrix is incremented by a real value $\alpha$. The increment $\alpha$ is typically in the range 0.01 to 0.1, and can vary depending on the characteristics of the collection and on one's confidence in the co-relevance data. In general, the larger $\alpha$ the more emphasis is placed on the co-relevance information and the greater the target similarity matrix will deviate from the baseline matrix. The effect of adding $\alpha$ to the $i,j$'th entry of the similarity matrix is to require that those two documents be represented more similarly than they would have been based on simple term co-occurrence. Note that $\alpha$ is added to the $i,j$'th entry of the similarity matrix each time documents are co-relevant with a query. Thus, an entry can be augmented multiple times if two documents are frequently co-relevant. In addition, the diagonal $i,i$'th entry is also incremented each time document $i$ is relevant to a query. This ensures that a document is always at least as similar to itself in the target matrix as to any other document. Figure 2 illustrates the construction of the target inter-document similarity matrix using co-relevance data.

For the CISO database, $\alpha$ is 0.015, and for Cranfield $\alpha$ is 0.1. For the MED database, two values of $\alpha$ are reported: 0.05 and 0.2. $\alpha$ is smaller for the CISO database because the average number of increments to the CISO baseline matrix (based on the number of co-relevant documents for each training query) is much larger than that for the Cranfield and MED baseline matrices. We have therefore selected $\alpha$ so that the cumulative effect of the increments to these three baseline matrices is roughly the same. In particular, for
Figure 2: Generation of the target similarities from the original term-space representations and a training set of queries with relevance-tagged documents, assuming increment $\alpha = 0.2$.

In the CISI database, the average number of documents relevant to each training query is 41 documents, while that for Cranfield is only 8 documents and that for MED is 23. Since an increment is made to the baseline matrix for all pairs of co-relevant documents, there is an $n^2$ relationship between the number of increments to the baseline matrix and $n$, the number of documents relevant to a training query. The estimated number of increments to the CISI baseline matrix is 150,480, an order of magnitude larger than the number of increments to the Cranfield database (14,340 increments) or MED database (12,230 increments). The $\alpha$ for CISI is therefore an order of magnitude smaller than the $\alpha$'s for Cranfield and MED, resulting in a comparable cumulative increment to the three baseline matrices. A small number of $\alpha$ values near these reported values were examined, though none resulted in substantial improvement over the values chosen.

The final step is normalization of the similarity matrix. The purpose of normalization
is to make the target similarity between every document and itself equal to 1.0. This normalization is important to ensure that no bias — which would make frequently relevant documents longer, and therefore more likely to be retrieved, than less frequently relevant documents — is introduced into the representations. In addition to making the similarity between a document and itself equal to 1.0, the normalization also makes the similarities between all other pairs of documents no greater than 1.0. Therefore, the most similar document to any one document is itself. The normalization method is to divide each row and then each column by the square root of the diagonal element for that row or column. That is, all elements of row (or column) \( i \) are divided by the square root of the \( i \)'th element of the row (column). This normalization results in a similarity matrix with all diagonal elements equal to 1.0. The target inner product similarity for any document to itself will therefore be 1.0, thereby attempting to constrain all documents to be represented in the semantic space by unit length vectors. Note that this normalization has an interesting effect on the resultant target similarity matrix: The overall effect has been to make documents which are not co-relevant to be slightly less similar and documents which are co-relevant to be slightly more similar, but the overall target document lengths remain identically 1.0. Although the MSM computation certainly will not yield unit length document vectors, the foregoing approach avoids a bias towards lengthening or shortening them.

We emphasize that the method used here to derive target inter-document similarity scores from relevance data is only one of a number of possibilities. For example, if a large amount of relevance data is available, it may not be necessary to seed the target matrix with the raw inter-document similarities; in addition, the normalization step might be omitted if the frequency of relevance of a document is to be reflected in its representation. Methods outside the MSM framework for incorporating relevance feedback in the document repre-
sentations have also been investigated, with considerable success (Brauen, 1971) (Wong, et. al., 1993). The selection of our method has been motivated primarily by the theoretical equivalence of Latent Semantic Indexing and the special case of MSM. The good performance of LSI relative to other competitive methods (Deerwester, et. al., 1990) suggests that inter-document similarities are a good starting point for the target similarity matrix. Furthermore, other research (e.g., Harman, 1986) suggests that normalization of the document representations can be quite helpful.

6 Results

The performance of the three indexing and retrieval algorithms, Metric Similarity Modeling (MSM), Latent Semantic Indexing (LSI), and SMART, is compared. Performance of the systems is compared for both the training queries and the test queries; the training queries are useful to illustrate the effects of MSM, while the test queries are used to estimate the expected performance benefit from the technique when used with novel queries.

Performance is measured by precision and recall levels averaged over the set of example queries. For ranked retrieval systems, recall can be systematically varied between 0.0 (0% of relevant documents retrieved) and 1.0 (100% of relevant documents retrieved) and the level of precision can be measured at each recall level (for details, see van Rijsbergen, 1983). By averaging over all queries, a graph of the precision vs recall gives a concise portrait of the ability of the retrieval system to accurately rank relevant documents before irrelevant ones. In the presentation of results, performance is also averaged over the different partitions of the query set into training and test sets.
Figure 3: Performance of Metric Similarity Modeling (MSM) and two comparable indexing and retrieval methods, SMART and LSI, on the training query sets. Performance for two databases, MED and Cranfield, are shown. It is not surprising that performance for MSM is superior, since the relevant documents for each of the training queries are constrained to be clustered by the MSM method.

6.1 Training Set Performance

The effect of applying MSM is best illustrated by examining the performance of MSM on the training queries. Figure 3 displays precision and recall graphs for two of the databases, Cranfield and MED. It is immediately apparent that MSM is retrieving relevant documents far better than LSI or SMART. This is particularly true for the MED database, in which almost all relevant documents are retrieved before any irrelevant ones.

We should expect that the performance of MSM on the training set should be superior to the benchmarks. This is because MSM has represented documents so that documents co-relevant to the training queries are more highly similar than they would be based on simple term co-occurrence. Thus, MSM is enforcing a clustering of related documents; co-relevant documents are self-similar and will separate from irrelevant documents. As long as
a training query is represented near its corresponding cluster, it is likely to retrieve more relevant documents than the baseline methods.

The MED database is particularly useful in illustrating this point. MED is an atypical database, in that the 30 queries used for training and testing partition the collection of documents into disjoint sets. That is, every document is relevant to either one or none of the 30 queries; thus, there are 31 disjoint classes of documents, one per query and one “garbage” class for the documents that are relevant to no query. Based on term occurrence, it is apparent that the classes of documents do not form convenient clusters with respect to their queries; if they did, precision would be extremely high as the cluster of co-relevant documents would be retrieved ahead of all irrelevant documents.

The effect of MSM is to reconfigure the representations, so that the documents are explicitly clustered based on their co-relevance. For MED, MSM is able to tightly cluster the different classes, since there is no overlap between the co-relevant sets which would force some documents to be represented between two classes. As illustrated in Figure 3, the class clusters are almost perfect for MSM using $\alpha = 0.2$.

Of course, MED is artificial in that the documents are partitioned by the queries. Most collections and user queries are not so structured. More typical training set performance is illustrated in Figure 3 by the Cranfield database. MSM performance is still far superior to the benchmarks; however, no strict clustering is created. In fact, a clustering is not possible since many documents belong to more than one query class. Rather, co-relevant documents are simply made more similar than their base rates of term co-occurrence, but they remain related to other, non-co-relevant documents.
6.2 Test Set Performance

To test the effectiveness of MSM as an indexing and retrieval algorithm, MSM is compared to the benchmark methods using the test set of queries. This test provides an expectation of the level of performance benefit over the benchmark methods when the MSM representations are used to retrieve documents for novel queries.

Precision and recall performance for the three databases is depicted in Figure 4. For two of the databases, Cranfield and CISI, MSM outperforms both the LSI and SMART benchmarks. In summary, MSM precision is 10% better on average than LSI, and 28% better than SMART, on the CISI database. On the Cranfield database, MSM precision is 10% better on average than LSI, and 31% better than SMART. All results are statistically significant (with $p < 0.002$, where query partition is the random factor, and recall level and indexing method are the experimental factors). This indicates that MSM is able to usefully extract meaningful semantic relationships from the target similarity data. These relationships are exploited in the document representations to result in good generalization performance.

Performance on the MED database is less positive. MSM precision is 1% lower than LSI's precision on average, but still 36% higher than SMART performance. This is likely again the result of the artificial structure of the MED database. Since the documents that are relevant to the test queries are completely distinct from the documents that are relevant to the training queries, there are few useful semantic relationships to be learned from the training set that can transfer to the test set. Interestingly, even though performance on the training set is so significantly improved, the structure within the documents relevant to the test queries is not too adversely affected: The 1% decrease in performance relative to LSI is not statistically significant (with $p > 0.14$ when increment $\alpha = 0.2$, and $p > 0.43$ when
\[ \alpha = 0.05; \text{again, query partition is the random factor, and recall level and indexing method are the experimental factors}. \]

### 6.3 Benefit at High Recall

For methods which rely on matching terms between the query and documents (e.g., the SMART term space approach), it is often difficult to retrieve all relevant documents. This is because many relevant documents will not contain any query terms. This inability to retrieve relevant documents which do not contain any of the query terms results in apparently low precision at high levels of recall, because the remaining relevant documents are retrieved from the remaining set of non-matching documents in an essentially random order.

Neither Metric Similarity Modeling nor Latent Semantic Indexing rely on term matching to retrieve documents. Since retrieval is performed by measuring similarity in a semantic space, all documents are similar to a query to some degree. All documents can therefore be ordered with respect to relevance to the query, and important distinctions can be made even at the highest levels of recall. Figure 5 illustrates this feature: The improvement in precision gained by LSI and MSM compared to SMART is greatest for higher levels of recall, averaged over all queries in all training partitions.

Figure 5 suggests that there is a peak in the improvement of both MSM and LSI compared to SMART, but the peak occurs at different levels of average recall for different databases (0.4 for CISTI, and 0.6 for Cranfield). Interestingly, the recall performance of weighted term matching alone is similarly varied. For example, on the CISTI database, only approximately 10% of all relevant documents are actually retrieved by the SMART retrieval system, on average over all the queries. However, for the Cranfield database, the SMART system is able to retrieve 39% of all relevant documents. Thus, the characteristic peak in
Table 1: Comparison of average levels of inter-document association for pairs of co-relevant documents and for pairs of documents in which one is relevant, one is not, from the Cranfield collection. In support of the Cluster Hypothesis, co-relevant documents tend to be more highly associated than relevant/irrelevant pairs, regardless of the indexing method. However, MSM accentuates this distinction, for both the training and test set of queries. The Index of Separation is the mean, over the query set, of the difference between association means.

improvement occurs at recall levels above the maximum attainable recall level for SMART. This supports the hypothesis that the greatest benefit of MSM and LSI is in retrieving those documents that contain no query terms.

Also worth noting in Figure 5 is the apparent depression in performance at the very lowest levels of recall, for both LSI and MSM when compared to SMART. This suggests that these methods might be best employed specifically to retrieve documents which term-matching cannot identify, but not to replace term-matching at the lowest levels of recall.

6.4 On the Cluster Hypothesis

The Cluster Hypothesis posits that closely associated documents will tend to be relevant to the same queries (van Rijsbergen & Sparck Jones, 1973) (van Rijsgergen, 1983) (Voorhees, 1986). If the Cluster Hypothesis is generally true for a collection, then the association measure between documents is likely a good indicator of semantic relatedness. The Cluster Hypothesis will be valid to varying degree for a given document collection, depending on characteristics of the collection and of the set of queries. For example, one collection may be very well partitioned into sets of co-relevant documents, while another may not.
The application of MSM we have presented here can be conceptualized as a procedure for enforcing the Cluster Hypothesis for a set of documents. The association level of pairs of documents in MSM is constrained by training to be low or high depending on whether the documents tend to be co-relevant. In the extreme, this forces all documents related to a single query to be highly similar to each other—a particularly strong form of the Cluster Hypothesis. For the set of documents co-relevant to the training queries, this statement is true by definition since MSM explicitly forces co-relevant documents to be similar for these queries. What is perhaps most interesting is that these constraints on the training documents exhibit positive transfer to the co-relevant test documents as well. Table 1 illustrates this phenomenon, for the Cranfield collection. The first two columns of data in this table give the mean inter-document association level (using cosine similarity; for details, see van Rijsbergen, 1983, pp. 45-47) for all pairs of documents that are co-relevant to a query and for all pairs of documents that are not co-relevant to a query. A measure of the difference between these two values (column 3) indicates the degree to which the Cluster Hypothesis might be valid for the collection: The larger the difference (assuming reasonable distributions of associations) the more highly clustered co-relevant documents are relative to non-co-relevant documents. The first set of values illustrates the expected result for the training queries: MSM greatly increases the separation between co-relevant clusters of documents and unrelated documents. The second set of values illustrates the positive transfer of this clustering to the test set of queries. Co-relevant documents are again better separated from the unrelated documents, relative to the SMART and LSI benchmarks.
7 Discussion

In this section, we discuss the major strengths of the proposed method and highlight certain assumptions that restrict its applicability. Directions for extension and generalization of the method are identified.

Metric Similarity Modeling is an interesting indexing algorithm because it creates document representations which can reflect the actual semantic organization of documents, rather than simply reflecting the vagaries of term co-occurrence. As the experimental results in the previous section demonstrate, the MSM representation of documents can be utilized to improve retrieval precision, especially at the high levels of recall where term-matching methods are known to fail. Most importantly, the semantic organization of the documents can abstract important semantic relations which generalize to novel situations and new queries. These improvements are possible because the method models semantic associations explicitly in the document representations.

MSM has additional useful properties in addition to these performance benefits. Foremost, the MSM algorithm generates an explicit functional relationship between the basic features of the documents, i.e., the terms, and the semantic space. This allows analysis of a term’s contribution to a document’s representation, similar to the analysis proposed by the authors of Latent Semantic Indexing (Deerwester, et. al., 1990, p. 399). The functional relationship also allows an arbitrary new document or query, never seen by the system, to be mapped into the semantic space based on its terms. This is a critical feature in order for the method to generalize to novel queries and documents.

An additional value of the MSM approach is the link it establishes between the indexing problem and Multidimensional Scaling (MDS) techniques. MDS is a class of data analysis methods which represent objects (in our case, documents and queries) in a multidimensional
space such that inter-object similarity matches the “semantic similarity” between the objects. Linking the indexing problem to MDS highlights potentially valuable generalizations of MSM and indicates how a number of the assumptions made in MSM, which critically affect the method’s applicability, may be relaxed.

Two major assumptions are that the target similarity constraints used by MSM must be *metric* and *exhaustive*. These assumptions require that a precise target similarity for all pairs of documents be provided to the algorithm. For many kinds of semantic constraints, it may be difficult to formulate the information in metric form. For example, it was difficult to model the co-relevance associations used in our experiments as metric similarity values. Though the experiments demonstrate that the raw term-vector similarities can be used as an initial estimate and then perturbed with the co-relevance information, the application is not completely natural. There are numerous other possible semantic constraints which are equally difficult to interpret as metric constraints, such as citation relationships between documents (the assumption being that two documents related by citation may be more related than their term-vector similarities suggest), and possible document classification information which may be available (e.g., the WESTLAW legal database, Rose, 1991). Though we would like to increase the similarity between two associated documents, it may not be readily clear how.

Both the metric and exhaustive assumptions can be relaxed. If only a subset of all the pairwise similarities are available, for example, then the error measure in equation 1, section 2.2 can be appropriately modified. The squared error between target and actual similarity values can be restricted to the applicable subset of available similarity values. Such an error measure can be minimized using iterative function minimization methods, such as gradient descent (Press, et. al., 1988), rather than by the explicit method (using the
Singular Value Decomposition) used in this paper. Alternatives to the metric assumption are more plentiful, and depend on the characteristics of the available similarity constraints. One alternative which we are currently examining (Bartell, et. al., 1994a, 1994b) is to use ordinal constraints on the similarities between documents and queries. Ordinal constraints do not specify a specific target similarity as do metric constraints; rather, they specify an ordering over the set of inter-object similarities. For example, an ordinal constraint requires that a document $i$ be more similar to document $j$ than a third document $k$ is to $j$, but does not specify the exact target similarity between the documents. As Wong & Yao (1990) have argued, ordinal constraints may be a natural interpretation of document/query relevance assessments. Ordinal constraints have been successfully used to optimize system parameters in a number of applications: The similarity measure used in a vector space retrieval model has been optimized, and multiple retrieval systems have been automatically combined into a single retrieval system (Bartell, et. al., 1994a, 1994b).

Another assumption made by MSM is that the mapping from term-vectors to the semantic representation is linear. We emphasize that finding the best linear mapping from data to $k$ dimensional semantic representation strongly constrains the quality of the representation. That is, the class of linear functions may not be functionally expressive enough to relate the term-space representation of documents to the optimal semantic space representation (i.e., the representation which best models the target similarity data by its inner product similarities). Non-linear functions, on the other hand, have the potential to more accurately model the target semantic structure. Non-linear functions can, for example, use as evidence the co-occurrence of two terms in the same document as meaning more than the linear sum of the two individual word meanings. This could be quite useful, as phrases

---

5 The mathematical details of this limitation have been provided in section 2.2.1.
such as “home run” and “systems theory” highlight. A non-linear MSM mapping function could be learned by minimizing the error criterion with iterative methods, as before, though with greater difficulty due to the introduction of local minima in the error surface. Note also that the likely introduction of additional parameters to a non-linear mapping function can increase the susceptibility of the model to over-fitting the training data and therefore failing to generalize to novel queries.

A final critical assumption made by MSM is that target similarities are interpreted as inner product similarities between vectors. Using inner product similarity has less effect on the applicability of the technique than the other assumptions discussed, but does have important theoretical significance. It often will not affect the applicability of MSM because many alternative similarity metrics are essentially inner product in form, with some additional normalization of the compared vectors (Wang, et. al., 1992). It is therefore often possible to normalize the the document vectors before applying the method, and then work with inner products of the normalized vectors. This is the approach we have taken in the experiments reported in section 5.

Theoretically, the significance of the inner product measure remains heavily debated in psychology. It is unclear whether semantic similarity can be accurately represented in a space under measures such as inner product similarity or euclidean distance (Nosofsky, 1991) (Tversky & Gati, 1978). For example, by the definition of inner product similarity, the similarity of $i$ to $j$ is equivalent to the similarity of $j$ to $i$. However, semantic similarity need not necessarily be symmetric. Tversky (1977) provides the example of two countries, China and North Korea. Human subjects reliably consider North Korea to be more similar to China than China is to North Korea. There are other axioms of metric spaces in addition to symmetry, and they also may be violated by arbitrary “real” semantic relationships.
These questions notwithstanding, multidimensional representational methods like MSM's have been quite useful in the modeling and interpretation of a large amount of psychological data. Though the document representations MSM will derive will be, at best, approximations to some underlying semantic structure in the document collection, we should expect that this approximation will be useful in determining relevance in the IR task. This hypothesis has been supported by the experiments.

8 Conclusion

We have proposed a method, Metric Similarity Modeling (MSM), for indexing documents based on modeling explicit target inter-document similarity scores. Documents are represented as vectors in a semantic space of low dimension (relative to an original term space representation). The vectors are chosen so that the similarities between the vectors most closely match the target similarities (in a least squares sense). The target similarities signify semantic associations between documents; this results in document representations in which similar vectors more accurately identify related documents, and dissimilar vectors identify less related documents. It is shown that Latent Semantic Indexing, an alternative dimensionality reducing approach, is optimally solving a specific Metric Similarity Modeling problem, when the target similarities are exactly the inter-document term co-occurrence scores. We generalize this by updating the co-occurrence scores with relevance feedback before mapping into the reduced space. The method is applied to three text databases which are standard in the information retrieval literature. The generally favorable performance of the technique, when compared to a weighted vector space approach and to Latent Semantic Indexing, suggests interesting potential for methods which explicitly modify the retrieval system to meet document similarity constraints.
Acknowledgements: The authors thank James Bunch, Chris Cole, Dean Inada, and Bob Clarke for useful discussions, and the anonymous reviewers for their helpful comments. The first author was supported in portions of this work by Peregrine Systems, Inc., Carlsbad, Ca. Address all correspondence to the first author. Email: bbartell@cs.ucsd.edu Surface mail: Dept. of Computer Science & Engineering-0114, UC San Diego, La Jolla, CA 92093.

References


A Details of Solution

We prove here that \( W^{(k)} = M_k^T C X^+ \) is a critical point to the configuration error in equation (1). Consider again the configuration error, \( E(W) \):

\[
E(W) = \sum_{i=1}^d \sum_{j=1}^d (\hat{S}_{i,j} - S_{i,j})^2
\]

\[
= \|\hat{S} - S\|_F^2
\]

\[
= Tr\{(\hat{S} - S)(\hat{S} - S)^T\}
\]

\[
= Tr\{(WX)^T WX(WX)^T WX
\]

\[-2S(WX)^T WX + SS\}

where \( Tr\{Z\} \) is the trace of a matrix \( Z \) computed by the sum of the diagonal entries of \( Z \), and \( Tr\{ZZ^T\} \) is equivalent by definition to the squared Frobenius norm \( \|Z\|_F^2 \). Calculating \( \partial E(W)/\partial W \) and setting this partial to 0, we find that any critical point \( \hat{W} \) of \( E(W) \) must satisfy

\[
WX(WX)^T WXX^T = WXSX^T
\]

(12)

We now demonstrate that a matrix \( \hat{W} = RM_k^T C X^+ \) is a solution to the equality in equation (12), where \( R \) is any full-rank \( k \times k \) rotation matrix (i.e. \( R^T R = I \)). By letting \( R = I \), we will have demonstrated that our solution for \( W^{(k)} \), i.e., \( W^{(k)} = M_k^T C X^+ \), is a critical point of the configuration error.

Consider first the left side of (12). Recall that the SVD of \( X \) is \( X = ULa^T \), \( S \) is decomposable into \( C^T C \), and \( M \Sigma N^T \) is given by the SVD of \( C A A^T \). Letting \( P^x \) be a
projection matrix derived from $X$ by $P^x = AA^T$, we can solve for the left side of (12):

$$
(WX)(WX)^T(WX)X^T = (RM_k^T CP^x)(P^xC^T M_k R^T)(RM_k^T CP^x)X^T
$$

$$
= RM_k^T M \Sigma N^T C^T M_k M_k^T M \Sigma N^T X^T
$$

$$
= R \Sigma_k N^T C^T M_k \Sigma_k N^T X^T
$$

(13)

These derivations make use of the facts that $X^+X = P^x$, $P^xP^x = P^x$, $R^TR = I$, and $CP^x = M \Sigma N^T$ by definition.

Now consider the right side of equation (12):

$$
WXSX^T = RM_k^T CP^x C^T CX^T
$$

$$
= RM_k^T M \Sigma N^T C^T CX^T
$$

$$
= R \Sigma_k N^T C^T CP^x X^T
$$

$$
= R \Sigma_k N^T C^T M \Sigma N^T X^T
$$

$$
= R \Sigma_k N^T C^T (M_{k,t\times r} + M_{r-k,t\times r}) \Sigma N^T X^T
$$

(14)

where $M_{k,t\times r}$ denotes the matrix $M$ retaining the first $k$ orthonormal columns, but with the other columns zero. $M_{r-k,t\times r}$ similarly denotes the matrix $M$ retaining the last $r-k$ orthonormal columns, but with the first $k$ columns zero. Obviously, $M = M_{k,t\times r} + M_{r-k,t\times r}$.

The derivation in (14) makes use of $X^T = P^x X^T$.

Comparing the final expressions derived in equations (13) and (14), we see that to verify the equality in equation (12) for $\hat{W} = RM_k^T C X^+$, we need only show that $R \Sigma_k N^T C^T M_{r-k,t\times r} \Sigma N^T X^T = \ldots$
0. Noting that $N^T C^T = \Sigma M^T$,

$$R \Sigma_k N^T C^T M_{r-k,t \times r} \Sigma N^T X^T$$

$$= R \Sigma_k \Sigma_k M_k^T M_{r-k,t \times r} \Sigma N^T X^T$$

$$= 0$$

since $M_k^T M_{r-k,t \times r} = 0$. Thus, $\hat{W} = R M_k^T C X^+$ is indeed a solution, and $W^{(k)} = M_k^T C X^+$ is a critical point of the configuration error.
Figure 4: Performance of Metric Similarity Modeling (MSM), SMART, and LSI, on the novel set of queries. MSM has successfully generalized on both the CISI and Cranfield databases. MSM does not generalize well on the MED test queries, due to the artificial characteristics of the database.
Figure 5: The improvement in precision relative to SMART’s precision varies significantly by recall level. Both the MSM and the LSI retrieval methods have greatest improvement at higher levels of recall. At the high recall levels, performance for retrieval methods like SMART is poor because no query terms occur in the relevant documents. The indicated improvement is averaged over all test-set partitions; error bars indicate one standard deviation from the mean.