

CS-1998-11

**A Comparison of Two Corpus-Based
Methods for
Translingual Information Retrieval**

Michael L. Littman Fan Jiang

Department of Computer Science
Duke University
Durham, North Carolina 27708-0129

June 29, 1998

A Comparison of Two Corpus-Based Methods for Translingual Information Retrieval

Michael L. Littman and Fan Jiang
Department of Computer Science
Duke University
Durham, NC 27708-0129
fax: 919-660-6519
{mlittman,fan}@cs.duke.edu

June 29, 1998

Abstract

In translingual information retrieval (TIR), ad hoc queries in any of a set of languages can be used to retrieve documents in any of a set of languages. Classical information-retrieval methods such as the vector-space model cannot be applied to TIR because they base similarity on the overlap of terms between queries and documents—this is typically zero in TIR. The generalized vector-space model (GVSM) and latent semantic indexing (LSI) are two variations of the vector-space model that make comparisons outside of term space. For this reason, both can be and have been applied to TIR. In this paper, we report on a series of experiments comparing the performance of GVSM and LSI on monolingual and translingual retrieval tasks. We find that the performance of both methods depends crucially on parameter settings, that LSI performs better, and that GVSM runs more quickly.

1 Introduction

Translingual information retrieval (TIR) is the problem of using ad hoc queries in any of a set of languages to retrieve documents in any of a set of languages. Classical information-retrieval methods, such as the vector-space model (Salton & McGill 1983), cannot be applied to TIR because similarity is based on the overlap of terms between queries and documents—this is typically zero in TIR.¹ Nevertheless, TIR is an important problem in our increasingly global information economy, and a better understanding of powerful, general TIR methods is needed.

The generalized vector-space model (GVSM) and latent semantic indexing (LSI) are two variations of the vector-space model that make comparisons outside of term space. For this reason, both can be and have been applied to TIR when a multilingual document-aligned corpus is available for training. These methods have been studied and compared (Dumais *et al.* 1997; Carbonell *et al.* 1997; Yang *et al.* 1997), although the tradeoffs in using these methods are still somewhat unclear.

In this paper, we build on the experiments of Carbonell *et al.* (1997) and Yang *et al.* (1997) to provide a more detailed picture of the translingual application of GVSM and LSI. Our main goals are to examine the effect of representation dimensionality on the outcome of the retrieval experiments (Section 3.3) and to fix an error in their implementation of LSI’s comparison formula (Section 3.5). However, we also found that, even replicating earlier experimental conditions as closely as possible, our results are strikingly different, with LSI outperforming all competing methods; we are not certain as to why our results are so different (Section 4).

We do not view this as a competition between LSI and GVSM; we feel that the techniques have complementary strengths and weaknesses. Our intent is simply to provide a clearer picture of the tradeoffs between the two approaches. Briefly, we find that LSI is better justified mathematically and achieves superior performance in some small-scale experiments (Section 3). On the other hand, GVSM is conceptually simpler, easier to implement, and presents less computational burden in preprocessing (Section 3.2). Therefore, the choice of method depends critically on the requirements of the application at hand.

2 Corpus-Based Translingual Information Retrieval

GVSM and LSI are two closely related techniques for automatic, corpus-based translingual information retrieval, both of which make use of a multilingual document-aligned corpus for training. In this section, we describe how these methods work mathematically. Our description closely follows that of Yang *et al.* (1997).

We describe the methods in terms of an English-Spanish bilingual example, although they have been applied to other language pairs including English-Greek (Berry & Young 1995), English-French (Landauer & Littman 1990), and English-Japanese (Landauer, Littman, &

¹An interesting refutation of this claim is given by Buckley *et al.* (1997), who show how the vector-space model can be applied with remarkable effectiveness in French-English TIR by treating English as potentially misspelled French.

Stornetta 1992), and even language triples such as English-French-Spanish (Littman & Keim 1997) and English-French-German (Rehder *et al.* 1997).

2.1 Definitions

The matrix E is an $m_E \times n$ term-by-document matrix of English documents, and the matrix S is an $m_S \times n$ term-by-document matrix of Spanish documents. These two document collections are parallel: document i in E and document i in S are presumed to be on the same topic.

Entries of the matrices are weighted and normalized; in our experiments, we used SMART’s ntc.ntc weighting:

$$E_{i,j} \propto c_{i,j}(\log(n+1) - \log(df_i)), \quad (1)$$

where $c_{i,j}$ is the number of times term i appears in document j , df_i is the number of documents in which term i appears, and the columns of E are normalized so that their vector lengths are 1. Other weighting methods are explored by Yang *et al.* (1997).

For ease of exposition, we consider only the case of using an English query to retrieve Spanish documents, however, the same method works for comparing English, Spanish, or mixed queries with English, Spanish, or mixed documents.

Retrieval works by ranking the similarity of the query to each document in the test collection. Let q be an m_E column vector representing the terms in the query, and d be an m_S column vector representing the terms in the document it is being compared to. Here, q_i represents the number of occurrences of term i in the query, weighted according to Equation 1.

To measure the similarity between vectors v and w , we use the cosine between the vectors: $\cos(v, w) = (v^T w) / \sqrt{v^T v + w^T w}$.

In the vector-space model, the similarity between a query and a document depends solely on the degree of overlap between the vectors: $sim(q, d) = \cos(q, d)$. In this work, we tag each term with its source language, so this similarity score will always give a zero result when comparing English and Spanish. In GVSM and LSI, the comparison is mediated by other matrices, and this allows positive similarity scores between queries and documents in different languages.

2.2 GVSM

The similarity score used in GVSM comparing an English query q to a Spanish document d has the form:

$$sim(q, d) = \cos(E^T q, S^T d). \quad (2)$$

Recall that E is a set of English training documents and S is a semantically similar set of documents in Spanish.

Both vectors $E^T q$ and $S^T d$ are $m \times 1$, so this comparison is well defined. The interpretation of the comparison is essentially that we look at the degree of overlap between the query q and each of the English training documents in E , and then we look at the Spanish document d and its degree of overlap with each of the Spanish training documents in S .

Since E and S are parallel, similarity between $E^T q$ and $S^T d$ indicates that they are related to related documents; this is taken as evidence that q and d are related themselves. Roughly, we are representing the new text objects q and d in the space formed by the original training documents.

To reduce the computational effort and sometimes to improve retrieval performance, it is possible to use the technique of *sparsification*. The idea here is that before the similarity of the vectors $E^T q$ and $S^T d$ is computed, all but the k most influential (largest absolute value) elements are set to zero. In Section 3.3, we report on the effect of varying k on retrieval performance. In analogy to LSI, described next, we refer to the sparsification factor k as the “dimension” of the GVSM space, although this is clearly not an accurate term.

2.3 LSI

LSI exploits a similar intuition to that of GVSM—defining a retrieval space based on documents in the training set allows text to be represented in a language-independent way—but does a great deal more work to capitalize on it. LSI begins by gluing the rows of E to the rows of S , and then analyzing the resulting matrix with a singular value decomposition. This means that we write

$$\begin{bmatrix} E \\ S \end{bmatrix} \approx \begin{bmatrix} U_E^k \\ U_S^k \end{bmatrix} \Sigma^k (V^k)^T,$$

where k is a predefined *dimensionality* of the analysis, Σ^k is a $k \times k$ diagonal matrix, U_E^k is an $m_E \times k$ matrix of representation for the English terms, U_S^k is an $m_S \times k$ matrix of representation for the Spanish terms, and V^k is an $n \times k$ representation of the documents. Note that we omit the k superscript when describing a full-dimensional analysis (if $k = \min(m, n_E + n_S)$); in this case the approximation is exact.

Note that $\begin{bmatrix} U_E^k \\ U_S^k \end{bmatrix}$ and V^k are orthonormal, meaning

$$\left(\begin{bmatrix} U_E^k \\ U_S^k \end{bmatrix} \right)^T \begin{bmatrix} U_E^k \\ U_S^k \end{bmatrix} = (U_E^k)^T U_E^k + (U_S^k)^T U_S^k = I$$

and $(V^k)^T V^k = I$. A standard result from linear algebra states that this decomposition yields the best rank- k approximation to the original matrix (Berry, Dumais, & O’Brien 1995).

Given the results of the SVD, the similarity score of query q and document d is

$$\text{sim}(q, d) = \cos((U_E^k)^T q, (U_S^k)^T d). \quad (3)$$

Note that Equation 3 differs from the erroneous LSI similarity score described by Yang *et al.* (1997); this is explained in detail in Section 3.5.

The common justification for Equation 3 is that, when d and q are both documents in the training collection and maximum dimensionality is used, LSI reduces to the vector-space model.

3 Experimental Results

In this section we describe our experimental results for GVSM and LSI.

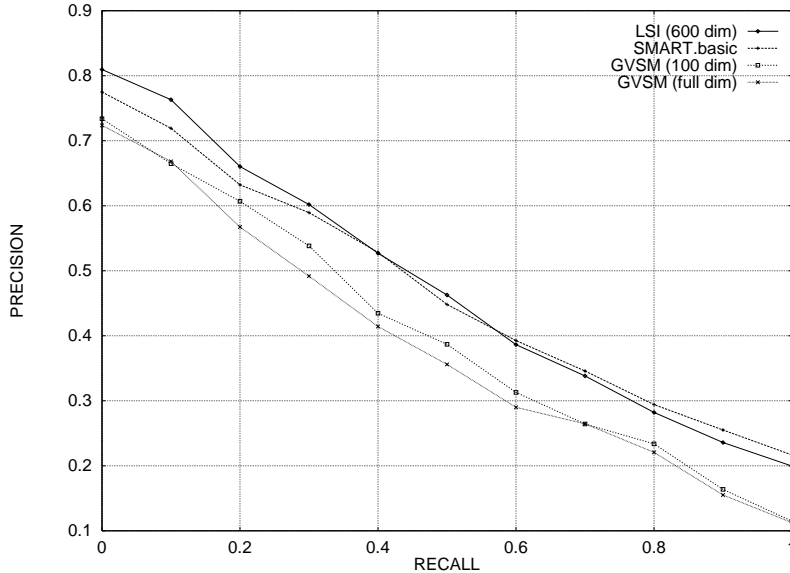


Figure 1: At the appropriate dimensionality, LSI outperforms SMART for monolingual information retrieval in the UNICEF collection.

3.1 Experimental Setup

The corpus used in the present study is the UNICEF collection created and described by Yang *et al.* (1997). This collection consists of 1134 training documents and 1121 testing documents, each in both English and Spanish. Yang *et al.* (1997) also prepared a set of 30 English queries along with exhaustive relevance judgments for these queries over all 1121 test documents. The relevance judgments were made between the English queries and the English test documents—relevance for the Spanish test documents were assumed to be the same, since these are translationally equivalent to the English.

Although the UNICEF collection is quite small by modern-day information-retrieval standards, the availability of relevance judgments makes it an invaluable resource. By way of comparison, the TREC-6 cross-language track collection contains hundreds of thousands of documents in each of three languages, but only 25 queries (“topics”) and non-exhaustive relevance judgments available for 22 of them at present.

We processed the training documents by first stemming each English and Spanish term using the SMART software available at Cornell’s ftp site (<ftp://cs.cornell.edu/pub/smart>) and using their English and Spanish stop-word lists. From this, we created matrices S and E , weighted according to Equation 1. We analyzed the joint English-Spanish matrix using a sparse SVD package. For each retrieval method we studied, we then compared each of the 30 queries against each of the 1121 test documents and computed an average 11-point precision score. Note that retrieval results are relevance ranked instead of top ranked, and this tends to inflate average precision scores slightly.

Figure 1 shows a standard recall-precision plot for the UNICEF collection, comparing the vector-space model (SMART), GVSM and LSI. These results are examined closely in Section 3.3.

3.2 Computational Effort

GVSM and LSI differ with regards to the amount of computational effort that is needed for training (preprocessing) and testing (retrieval). We present here rough complexity analyses and empirical results. As our implementations have not been heavily optimized for either method, we present empirical results only to be suggestive of the differences in computational demands between the methods.

For preprocessing, both methods begin by creating the appropriate term-by-document matrices. This work is identical for both methods; roughly $O(N)$ or linear in the number N of non-zero entries in the matrices. On the UNICEF collection, our implementation takes about 5 minutes to create the necessary matrices.

At this point, LSI performs a singular value decomposition. Empirically, this seems to run in time $O(kN)$ —the number of dimensions times the number of non-zero entries in the matrices—but with a very large constant in the “big Oh.” This can amount to a substantial amount of processing; Rehder *et al.* (1997) report that the analysis took 2.5 days on a powerful workstation for 800 dimensions, 240k terms, and 40k documents. For the UNICEF collection, however, LSI’s preprocessing time is considerably less—approximately 20 minutes for a full-dimensional analysis. For much larger collections, it might be necessary to train using a small collection and then use fold-in and updating techniques developed for monolingual LSI (Berry, Dumais, & O’Brien 1995).

One final step of preprocessing can be performed for both LSI and GVSM. For each of the n' documents in the test collection, a transformed vector representation can be computed containing k non-zero elements per document. Let l be the expected number of terms in a document (or query). For GVSM, a document representation is computed by transforming the document by the training matrix in $O(nl)$, then sparsifying it in $O(n)$. This is carried out for every testing document, resulting in a total cost of $O(n'nl)$.

For LSI, the representation of the testing documents involves adding together the k -dimensional vector representations for all the terms in the document, resulting in a total of $O(n'kl)$. As $k \leq n$, this will typically be less than the work performed by GVSM.

To match a query against the set of testing documents, the query is first transformed to its vector representation ($O(kl)$ for LSI and $O(nl)$ for GVSM), then each document vector is compared to the resulting query vector by computing a cosine. This takes $O(n'k)$ for both methods, although the computation itself is quite different—GVSM is an n -dimensional sparse vector-vector multiplication, whereas LSI is a k -dimensional dense vector-vector multiplication.

The times to complete retrieval tests for all 30 queries in the UNICEF collection range from about 30 seconds for $k = 200$ to about 1 minute for $k = 1134$. Running times for GVSM and LSI do not appear to vary significantly for the same k .

In comparing the computational effort required by GVSM and LSI, it is important to separate out the one-time preprocessing costs from the incremental cost of processing single queries. For preprocessing, GVSM is clearly preferable, running several times faster on the UNICEF collection; tests on other, larger, collections are needed to better quantify this difference. For individual query processing, LSI and GVSM are similar given similar dimensionalities; however, there is no *a priori* reason to believe that the two methods are

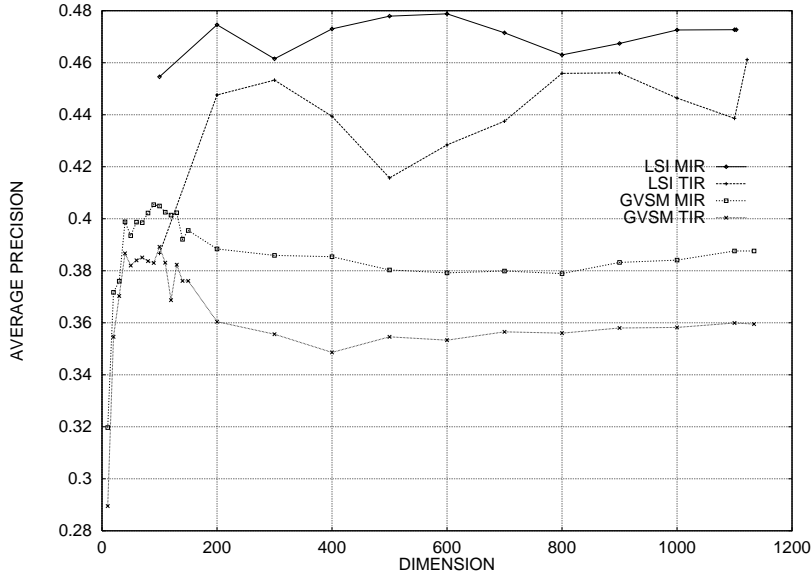


Figure 2: As the number of dimensions (sparsification in the case of GVSM) is varied, the performance of LSI and GVSM on MIR and TIR changes quite a bit.

	Yang <i>et al.</i> (1997)	dim	Figure 2	dim	Max Performance	dim	Max Dimensions	dim
LSI MIR	.3689	200	.4746	200	.4788	600	.4727	1103
LSI TIR	.3093	200	.4476	200	.4612	1122	.4612	1122
GVSM MIR	.4008	200	.3884	200	.4054	90	.3876	1134
GVSM TIR	.3804	200	.3604	200	.3892	100	.3595	1134

Table 1: Our experimental results differ from those reported in earlier work.

comparable when matched for dimensionality.

It is clear that the computational costs hinge critically on the number of dimensions needed by each method to achieve good performance; although such a formula remains unknown, the next section provides information on how the number of dimensions affects retrieval performance for the UNICEF collection.

3.3 Varying Dimensions

In this section, we examine the effect of varying the number of dimensions on monolingual and translingual retrieval performance. Our results appear in Figure 2 and Table 1. For LSI, we tested dimensionalities from 100 to 1100 in intervals of 100, the maximum dimensionality, and additionally in the case of GVSM, dimensionalities from 10 to 150 in intervals of 10.

There are several noteworthy features of this graph.

For GVSM, performance decreases gradually as k is decreased from its maximum value of 1134 (no sparsification) to around 500. Surprisingly, however, performance then begins to climb with falling dimension. For TIR, the peak value of .3892 appears at 100 and is a full 8% larger than the performance with no sparsification; thus, not only does sparsification

improve retrieval efficiency, it can also improve retrieval performance.

The performance of our implementation of GVSM at $k = 200$ is .3604, which is quite a bit different from the value reported by Yang *et al.* (1997) of .3804 (Table 1). This is puzzling, as we controlled for differences in preprocessing, weighting, training set, and sparsification value in our experiments. We are continuing to pursue possible sources of error in our code. Nonetheless, our peak value for TIR GVSM’s performance is .3892 at sparsification $k = 100$, which is larger than that reported by Yang *et al.* (1997). The same holds true of our MIR performance at $k = 200$ —we obtained .3884 instead of .4008, although our GVSM’s performance at optimal sparsification $k = 90$ is .4054.

The performance of LSI in these experiments is a good bit better than that of GVSM. LSI’s monolingual performance is .4727 at the maximum dimension of $k = 1103$,² which, as predicted, is nearly identical to that achieved by the vector-space model (SMART.basic) given by Yang *et al.* (1997) (Table 1) and in our own experiments. However, LSI’s peak MIR performance of .4769 occurs at $k = 600$. Over the entire range of dimensions we studied, LSI’s monolingual performance is strong.

LSI’s translingual performance was more erratic, varying from a low of .3867 at $k = 100$ to a high of .4612 at $k = 1122$ (maximum dimensionality) with several peaks and valleys in between. In spite of the massive variability, LSI’s TIR performance dominates GVSM’s TIR and MIR performance for all tested values of $k > 100$. Comparing LSI’s peak TIR performance to the best known MIR performance (also LSI, but for a different dimensionality), we get only a 4% decrease in performance when moving from monolingual to translingual retrieval.

Of the 12 values of k we tested, the performance of LSI for 9 of them dominates the best TIR method described by Yang *et al.* (1997). It is interesting to note that, as earlier studies have suggested, large dimensionalities appear to work better than small ones for LSI in the TIR setting. The observation needs to be studied more carefully in larger collections.

An important observation from these experiments is that the performance of LSI and GVSM varies substantially by dimensionality. Algorithms for parameter tuning are needed for both methods so that they can be used at maximum efficiency without human intervention.

3.4 Mate Retrieval

Early experiments on the use of LSI for translingual information retrieval used *mate retrieval* as a measure of performance (Landauer & Littman 1990). The mate-retrieval task is simple because it only requires a document-aligned testing corpus; no queries and relevance judgments are needed. However, it is not very closely related to realistic applications, so its overall usefulness is somewhat in doubt.

The mate-retrieval task is defined as follows. Begin with a document-aligned collection of English and Spanish training documents. Take an English document, compute its similarity to all Spanish documents and sort the resulting list. Now find the Spanish “mate” to which

²Apparently, the E matrix for this collection is rank deficient. Our SVD code could only find 1103 of the 1122 possible non-zero singular values in the English-only case.

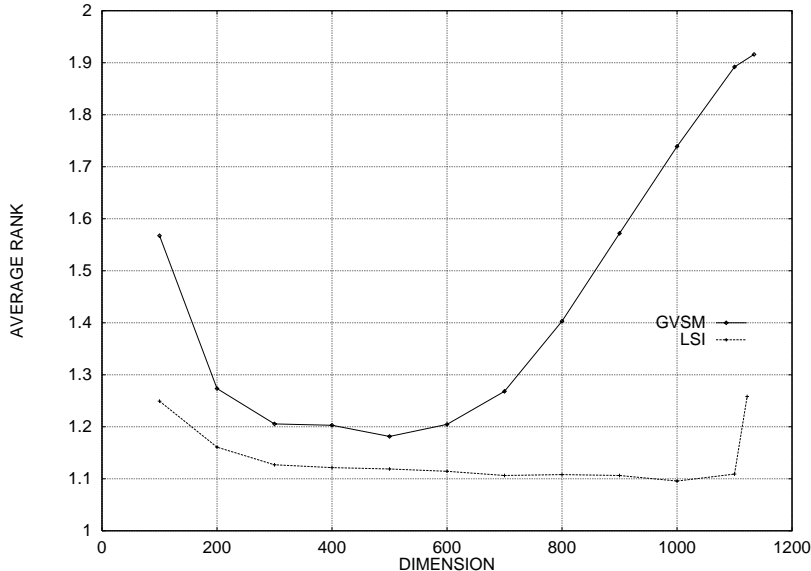


Figure 3: Compared to average precision performance, the mate-retrieval performance of GVSM and LSI varies differently with the number of dimensions (sparsification in the case of GVSM). Here, lower scores are better.

the English document is aligned and note its rank in the sorted list. Finally, average this rank over all English test documents.

Figure 3 presents the results of mate retrieval on the UNICEF corpus. It illustrates a number of important properties of the mate-retrieval task for this corpus.³ First, it is able to correctly predict the relative performance between GVSM and LSI. However, it is not at all able to predict the relative performance of either GVSM and LSI as a function of dimension. In fact, for GVSM, mate-retrieval performance drops dramatically from $k = 400$ to $k = 100$ just as average precision improves sharply. Similarly, mate-retrieval performance for LSI improves steadily with increasing dimension, whereas average precision is much more erratic.

On the basis of these results, the utility of mate retrieval appears mixed.

3.5 Singular Values

Consider the following comparison equation, which generalizes Equation 3,

$$\text{sim}(q, d) = \cos((\Sigma)^r (U_E)^T q, (\Sigma)^r (U_S)^T d). \quad (4)$$

Using different values of r , this equation allows us to include different numbers of multiples of the singular value matrix Σ in the comparison.

Yang *et al.* (1997) described LSI as using the comparison in Equation 4 with $r = -1$. In fact, the LSI comparison uses $r = 0$, i.e., no multiples of the singular values are included.

³Note that we excluded two of the test-document pairs that were clearly misaligned in the collection. These document pairs were identified by running the mate-retrieval test—an unanticipated side benefit of this testing method.

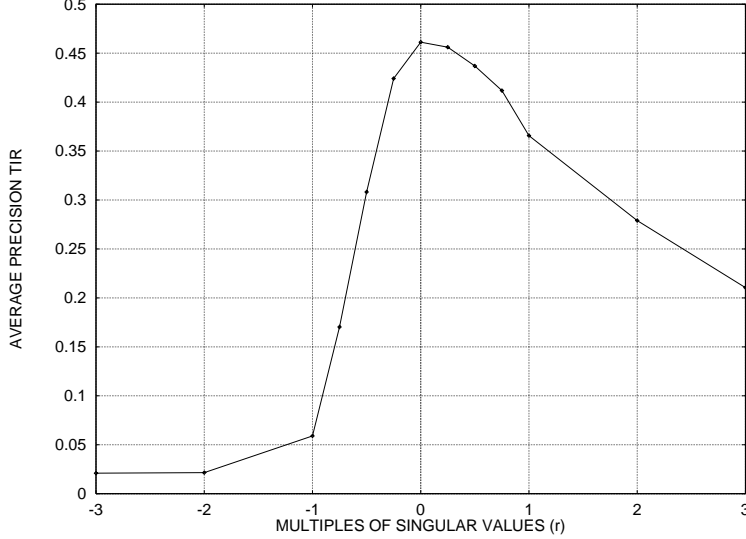


Figure 4: Varying the number of multiples of the singular-value matrix in the similarity computation dramatically effects TIR performance. When $r = 0$, the comparison is equivalent to full-dimensional LSI and when $r = 1$ it is full-dimensional GVSM.

The reason for this error is simple: comparisons in LSI are often described in terms of the creation of “pseudo-documents” (Deerwester *et al.* 1990), which involve multiplication by Σ^{-1} . However, comparisons between pseudo-documents include a factor of Σ , which cancels out the Σ^{-1} . This second point is not made clearly by Berry, Dumais, & O’Brien (1995), and in some other LSI papers.

It is interesting to note that the similarity score used in full-dimensional GVSM (no sparsification) can be expressed by Equation 4 with $r = 1$. To see this, note that for GVSM

$$\begin{aligned}
sim(q, d) &= \cos(E^T q, S^T d) \\
&= ((E^T q)^T (S^T d)) / \sqrt{(E^T q)^T (E^T q) + (S^T d)^T (S^T d)} \\
&= (q^T E S^T d) / \sqrt{q^T E E^T q + d^T S S^T d} \\
&= (q^T U_E \Sigma V^T V \Sigma (U_S)^T d) / \sqrt{q^T U_E \Sigma V^T V \Sigma (U_E)^T q + d^T U_S \Sigma V^T V \Sigma (U_S)^T d} \\
&= (q^T U_E \Sigma \Sigma (U_S)^T d) / \sqrt{q^T U_E \Sigma \Sigma (U_E)^T q + d^T U_S \Sigma \Sigma (U_S)^T d} \\
&= \cos(\Sigma (U_E)^T q, \Sigma (U_S)^T d)
\end{aligned}$$

Given that both GVSM and LSI can be viewed as parameterized variations on the same equation, it is interesting to ask how the performance of the algorithm varies as a function of this parameter. This is given in Figure 4, which shows that the peak performance is, in fact, attained with the LSI similarity score and is less for GVSM and much less for the LSI similarity equation described by Yang *et al.* (1997).

As a form of verification, we compared the performance obtained using the similarity equations for GVSM (no sparsification—Equation 2) to that of LSI (full dimension) with a single multiple of the singular values. As predicted by the equation above, these two methods give the same performance of approximately .36, as is visible in Figures 2 and 4.

4 Conclusions

We described a set of experiments comparing GVSM and LSI for translingual information retrieval. Our experiments built on those of Carbonell *et al.* (1997) and Yang *et al.* (1997), providing additional data on the effect of dimension reduction and the importance of the use of the correct number of multiples of the singular values in LSI’s similarity formula.

Our performance results for LSI are substantially better than those reported by Yang *et al.* (1997), in part because of problems in their implementation of LSI’s similarity formula. However, this does not serve as a complete explanation, as our experimental results for the performance of GVSM differ from theirs also.

There are two factors we can think of that might have led to this difference. One is in the precise time in the processing order in which the matrices are normalized. For example, are the lengths of document vectors in the E matrix normalized to 1 before the SVD is carried out? In our experiments, they were. Also, in our experiments, cross-language homonyms such as names and numbers were kept separate; it would probably improve performance to allow such terms to be given a single representation in the two languages. We intend to try to experiment with these variations to see how they affect performance.

Overall, our results indicate that larger scale experiments are crucial. We are beginning to look at the data from the TREC-6 cross-lingual retrieval track (Rehder *et al.* 1997), which consists of a large English-French-German trilingual training set (roughly 80k documents) and considerably larger non-parallel monolingual test collections (between 100k and 200k documents, depending on the language). The behavior of various algorithms when scaling up to larger collections is a critical step in assessing the practicality of these approaches.

Another important direction to move is to reduce the dependence on document-aligned training corpora. We hope to use experiments such as the ones reported here as baselines for studying more sophisticated algorithms.

For the UNICEF collection, we find LSI to dominate GVSM in performance, and GVSM to dominate LSI in preprocessing computational efficiency. An interesting direction for future work is to examine other algorithms that might make this tradeoff differently—spending a small amount of additional computation to boost performance.

References

- Berry, M. W., and Young, P. G. 1995. Using Latent Semantic Indexing for multilanguage information retrieval. *Computers and the Humanities* 29(6):413–429.
- Berry, M. W.; Dumais, S. T.; and O’Brien, G. W. 1995. Using linear algebra for intelligent information retrieval. *SIAM Review* 37(4):573–595.
- Buckey, C.; Mitra, M.; Walz, J.; and Cardie, C. 1997. Using clustering and SuperConcepts within SMART: TREC 6. In *The Sixth Text Retrieval Conference Notebook Papers (TREC6)*. National Institute of Standards and Technology Special Publication. 1–18.
- Carbonell, J.; Yang, Y.; Frederking, R.; Brown, R. D.; Geng, Y.; and Lee, D. 1997. Translingual information retrieval: A comparative evaluation. In *Proceedings of Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97)*.

- Deerwester, S.; Dumais, S. T.; Furnas, G. W.; Landauer, T. K.; and Harshman, R. A. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science* 41(6):391–407.
- Dumais, S. T.; Letsche, T. A.; Littman, M. L.; and Landauer, T. K. 1997. Automatic cross-language retrieval using latent semantic indexing. In Hull, D., and Oard, D., eds., *Cross-Language Text and Speech Retrieval: Papers from the 1997 AAAI Spring Symposium*. The AAAI Press.
- Landauer, T. K., and Littman, M. L. 1990. Fully automatic cross-language document retrieval using latent semantic indexing. In *Proceedings of the Sixth Annual Conference of the UW Centre for the New Oxford English Dictionary and Text Research*. 31–38.
- Landauer, T. K.; Littman, M. L.; and Stornetta, W. S. 1992. A statistical method for cross-language information retrieval. Unpublished manuscript.
- Littman, M. L., and Keim, G. A. 1997. Cross-language text retrieval with three languages. Technical Report CS-1997-16, Department of Computer Science, Duke University. Presented at Cross-lingual Information Retrieval SIGIR 97 Workshop.
- Rehder, B.; Littman, M. L.; Dumais, S.; and Landauer, T. K. 1997. Automatic 3-language cross-language information retrieval with latent semantic indexing. In *The Sixth Text Retrieval Conference Notebook Papers (TREC6)*, 103–110. National Institute of Standards and Technology Special Publication.
- Salton, G., and McGill, M. J. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill.
- Yang, Y.; Brown, R. D.; Frederking, R. E.; Carbonell, J. G.; Geng, Y.; and Lee, D. 1997. Bilingual-corpus based approaches to translingual information retrieval. In *The 2nd Workshop on “Multilinguality in Software Industry: The AI Contribution (MULSAIC’97)”*. At URL <http://www.iit.nrcps.ariadne-t.gr/%7ecostass/mulsaic97.html>.