

# Document Categorization Using Latent Semantic Indexing

## Abstract

The purpose of this research is to develop systems that can reliably categorize documents using the Latent Semantic Indexing (LSI) technology [2]. Initial research has indicated that the LSI technology shows great promise in constructing categorization systems that require minimal setup and training. Categorization systems based on the LSI technology do not rely on auxiliary structures (thesauri, dictionaries, etc.) and are independent of the native language being categorized (given the documents can be represented in the UNICODE character set).

Three factors led us to undertake an assessment of LSI for categorization applications. First, LSI has been shown to provide superior performance to other information retrieval techniques in a number of controlled tests [3]. Second, a number of experiments have demonstrated a remarkable similarity between LSI and the fundamental aspects of the human processing of language [6]. Third, LSI is immune to the nuances of the language being categorized, thereby facilitating the rapid construction of multilingual categorization systems.

The emergence of the World Wide Web has led to a tremendous growth in the volume of text documents available to the open source community (e.g., special interest web pages, digital libraries, subscription news sources, and company-wide Intranets). Quite coincidentally, this has led to an equally explosive interest in accurate methods to filter, categorize and retrieve information relevant to the end consumer. Of special emphasis in such systems is

the need to reduce the burden on the end consumer and minimize the system administration of the system.

We will describe the implementation of two successfully deployed systems employing the LSI technology for information filtering (English and Spanish language documents) and document categorization (Arabic language documents). The systems utilize in-house developed tools for constructing and publishing LSI categorization spaces. Various interfaces (e.g., SOAP-based Web service, workflow interfaces, etc.) have been developed that allow the LSI categorization capability to address a variety of customer system configurations. The core LSI technology has been implemented in a modern J2EE based architecture facilitating its deployment on a variety of platforms and operating systems. We will describe some early results on the accuracy and use of the systems.

## Introduction

Latent Semantic Indexing is an automated technique for the processing of textual material. It provides state-of-the-art capabilities for:

- automatic document categorization;
- conceptual information retrieval, and;
- cross-lingual information retrieval.

A key feature of LSI is that it is capable of automatically extracting the conceptual content of text items. With knowledge of their content, these items then can be treated in an intelligent manner. For example, documents can be routed to individuals based on their job responsibilities. Similarly, e-mails can be

filtered accurately. Information retrieval operations can be carried out based on the conceptual content of documents, not on the specific words that they contain. This is very useful when dealing with technical documents, particularly cross-disciplinary material.

LSI is not restricted to working with words; it can process arbitrary character strings. For example, tests with MEDLINE data have shown that it deals effectively with chemical names. Points in an LSI space can represent any object that can be expressed in terms of text. LSI has been used with great success in representing user interests and the expertise of individuals. As a result, it has been employed in applications as diverse as capturing customer preferences and assigning reviewers at technical conferences.

In cross-lingual applications, training documents from one language can be used to categorize documents in another language (for languages where a suitable parallel corpus exists). A discussion on LSI's cross-lingual capabilities can be found in [4].



## Training

Text categorization is the assignment of natural language texts to one or more predefined categories based on their content [1]. Text categorization systems run the gamut from those that employ trained professionals to categorize new items to those that are based on natural language clustering algorithms requiring no human intervention to guide the categorization process; the former process being very time consuming and costly, while the latter is the pinnacle of text categorization. Practical, cost-effective implementations fall somewhere in between.

Supervised text categorization has a learning (or training) component where pre-defined category labels are manually assigned to a set of documents that will become the basis for subsequent automated categorization. Text categorization systems performing unsupervised training (or learning) automatically detect clusters or other common themes in the data that identify topics or labels without manual labeling of the data.

When used in text categorization applications LSI requires a labeled training set of documents. Labeled training sets can be as few as seventy-five to several thousand in number. It is possible to use a small number of labeled documents to bootstrap the supervised learning process. After building an initial index with labeled test documents, additional documents can be submitted as queries, and query documents close in similarity to labeled documents in the index (within some pre-specified threshold value) can then be associated with the same label. In this manner the labeled test set can be grown over time with a significant reduction in the human effort required to build a large labeled test set.

When using smaller-sized training sets (less than 300-600 documents) LSI may require some additional tuning of the dimensionality of the categorization index to capture the higher ranked latent features in the training set. This is easily accomplished through a graphical user interface and iterations through re-indexing of the training set.

We have also found that adding unlabeled data ("background" text) in the presence of small labeled test sets improves the latent structure of the categorization index leading to improved accuracy. Similar results have been reported in the literature [7, 10]. Figure 1 shows the effect of background material on a small labeled test set of 300 documents. Unlabeled examples (e.g., web pages, emails, news stories) are much easier to locate and collect than labeled examples.

A common critique of LSI in the literature is the relatively high computational and memory requirements required by LSI to function [5]. However, with the advent of modern processors and their ever-increasing speeds this former consideration has been overcome. Training LSI with a moderate training set can be accomplished in a matter of minutes on current corporate desktop PCs with less than 1GB of memory. Larger sets of training documents require less than 10 minutes on equivalent PCs.

## Test Corpus and Performance Measures

LSI as a text categorization engine has been deployed in a number of real world applications as described below. To compare its performance to other published results we used the ModApte version of the Reuters-21578 test set [11]. The ModApte version has been used in a wide number of studies [8] due to the fact that unlabeled documents have been eliminated and categories have at least one document in



the training set and the test set. We followed the ModApte split defined in the Reuters-21578 data set in which 71% of the articles (6552 articles) are used as labeled training documents and 29% of the articles (2581 articles) are used to test the accuracy of category assignments.

Many different evaluation criteria have been used for evaluating the performance of categorization systems. For evaluating the effectiveness of category assignments to documents by LSI we adopted the break-even point (the arithmetic average of precision and recall) as reported in [1] and [8], and the total ('micro-averaged') precision P and recall R (defined in [9]). The micro-averaged break-even point is defined as  $(P+R)/2$ .

## Comparison with Other Techniques

Table 1 summarizes the global performance score for LSI along with the best performing classifier from [8]. As can be seen from Table 1, the LSI miF1 value was competitive with the miF1 value for the Support Vector Machine (SVM) in [8].

**Table 1. Performance summary**

Method	miR	miP	miF1	MaF1	Error
LSI	.8880	.8900	.8890	.5880	.0040
LSI	.8120	.9137	.8599	.5251	.0036
miR=micro-avg recall		miP=micro-avg prec			
miF1=micro-avg F1		maF1=macro-avg F1			

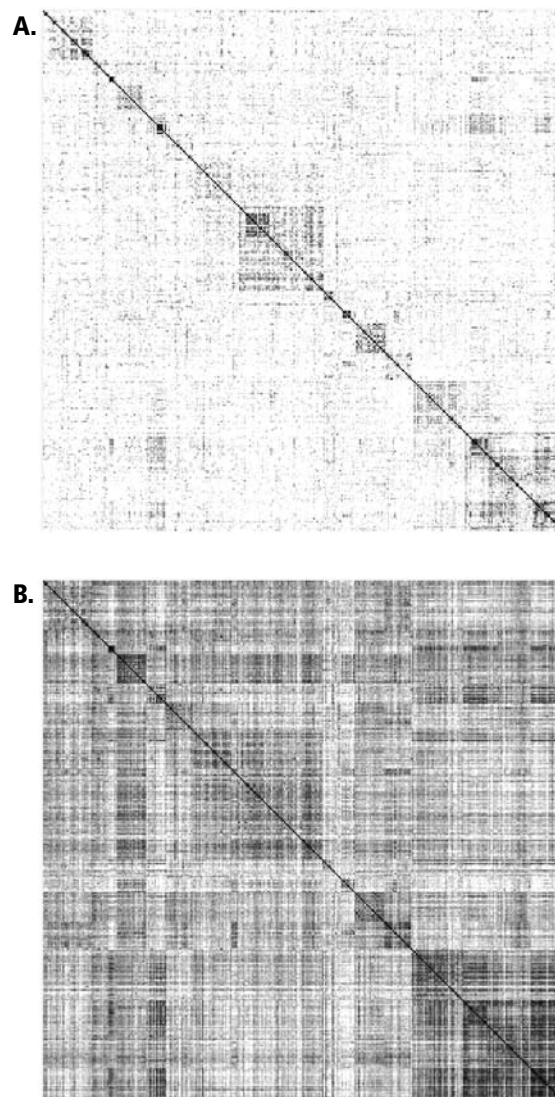
While the document counts between the two studies were not exactly the same the overall ratios of training set to test set were almost exactly the same; in [8] the ratios were 72 and 28 percent for the training and test sets, respectively. Additionally, in [8] there was an assumption that documents could fit into more than one category; unlabeled documents were eliminated and categories had to have at least one document in a training set and the test set. In [8] the number of categories per document was 1.3, on average. The category per document ratio for the ModApte data set used in this paper was one. This is a more stringent restriction on text categorization classifiers. The LSI results reported in Table 1 reflect this constraint.

In [1] the assumptions concerning what documents made up the ModApte split differ slightly from [8] and the test set used in this study. The mean number of categories per document for [1] was 1.2, but many documents were not assigned to any of the 118 categories, and some documents were assigned to 3 or more categories. The Support Vector Machine (SVM) was the most accurate text categorization method in [1] with an overall miF1 rating of 0.8700 placing it between the LSI and SVM results reported in Table 1.

## Real World Applications

### Information Filtering/Knowledge Discovery

In this application, the customer had a proprietary process for collecting English and Spanish content on a periodic basis. Once collected, the content was indexed with Boolean retrieval technology and made available to analysts for review. Analysts constructed and executed queries to retrieve content specific to their particular interests. Results varied depending on the expertise analysts possessed in constructing queries. An additional drawback was that analysts spent a large amount of their time searching for relevant content rather than analyzing content.



**Figure 1.** Graphic of the training space illustrating the effects of background material on the LSI training set. Figure 1a shows the similarity matrix for the training set. The row and column axes represent the documents in the training set; the diagonal shows that every document is related to itself. The stronger outlines surrounding the diagonal represent the labeled classes within the training data. Figure 1b shows how background material strengthens the latent relationships in the training data.

To address the above situation, LSI technology was integrated into the workflow to replace the Boolean retrieval technology. Rather than construct and execute queries analysts supplied representative content (i.e., documents) relevant to their areas of interest. This material was tagged, and indexed. Content collected on a periodic basis was compared to the index of analyst relevant content. Content similar in nature (within a specified threshold) to analyst content was routed to the appropriate analyst. Restructuring of the workflow in this manner resulted in a continuous push of relevant content to analysts, resulting in a significant increase in productivity on the part of the analyst. This system has been in production for over two years.

## Document Categorization/ Prioritization

In this application the customer had a high volume of Arabic language content and an insufficient number of Arabic-qualified analysts to review all the content. In order to ensure that relevant content was not overlooked all of the material had to be examined leading to overworked analysts and a situation where, potentially, some item of important material might be overlooked.

To address the above situation, a training set of Arabic content was constructed and

labeled according to customer-defined categories. The system was trained with the labeled training set. An additional 20,000 relevant Arabic documents were selected and used as background training material. Integration with the customer's workflow was accomplished using a SOAP-based web service. Arabic documents for categorization are passed to the web service. A ranked list of categories and associated similarity scores are sent back to the client process. Based on customer-defined rule sets, the client process makes decisions about the importance of the documents and their disposition. Highly ranked documents are immediately forwarded to analysts, less important documents are stored for later examination during periods of analyst workload, and uninteresting documents were discarded. This system has been in production for one year. During customer acceptance testing, this system demonstrated 97% accurate assignment of Arabic-language documents to individual categories. This result was measured using real-world documents with significant quantities of noise.

## Conclusions

The LSI technology has matured to the point where it is a particularly attractive approach for text categorization. Text categorization results with LSI are competitive, on comparable test sets, with the best results

reported in the literature. A definite advantage to the LSI text categorization technology is the native support for international languages.

## Lessons Learned

LSI categorization can perform well with very limited quantities of training data – generally with only a few examples per category. This is due, in great part, to the exceptional conceptual generalization capabilities of LSI.

User feedback can be incorporated to continually improve performance.

The LSI technique has a significant degree of inherent noise immunity with regard to errors in the documents being processed.

Documents can be assigned to multiple categories, with reliable indications of the degree of similarity to each category.

## Acknowledgements

We thank Roger Bradford, Janusz Wnek, and Rudy Keiser for their useful comments when reviewing this paper.

[1] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization, Proceedings of ACM-CIKM'98, 1998.

[2] S. Deerwester et al. Indexing by Latent Semantic Analysis, Journal of the Society for Information Science, 41(6), pp. 391-407, October, 1990.

[3] S. Dumais. Using LSI for Information Retrieval, Information Filtering, and Other Things, Cognitive Technology Workshop, April 4-5, 1997.

[4] S. Dumais et al. Automatic Cross-linguistic Information Retrieval using Latent Semantic Indexing, in SIGIR'96 - Workshop on Cross-Linguistic Information Retrieval, pp. 16-23, August 1996.

[5] G. Karypis and E. Han. Fast supervised dimensionality reduction algorithm with applications to document categorization and retrieval, Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management, 2000.

[6] T. Landauer and D. Lanham. Learning Human-like Knowledge by Singular Value Decomposition: a Progress Report, Advances in Neural Information Processing Systems 10, Cambridge: MIT Press, pp. 45-51, 1998.

[7] K. Nigam. Using unlabeled data to improve text classification, PhD. Thesis, Carnegie Mellon University, May 2001.

[8] Y. Yang and X. Liu. A re-examination of text categorization methods, Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99), pp. 42-49, 1999.

[9] Y. Yang. An evaluation of statistical approaches to text categorization, Journal of Information Retrieval, Volume 1, No. 1/2, pp. 67-88, 1999.

[10] S. Zelikovitz and H. Hirsh. Using LSI for Text Classification in the Presence of Background Text, Proceeding of CIKM-01, 10th ACM International Conference on Information and Knowledge Management, 2001.

[11] The Reuters-21578 collection is available at: <http://www.daviddlewis.com/resources/testcollection/reuters21578/>