CONF-9608120--1

# Information Fusion for Automatic Text Classification*

V. Dasigi

Department of Computer Science and Information Technology
Sacred Heart University
Fairfield, CT.

R. C. Mann
V. A. Protopopescu

Intelligent Systems Section
Computer Science and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, TN.

Paper Submitted To:

**Workshop on Foundations of Information/Decision Fusion: Applications to Engineering Problems**

Washington, DC
August 7-9, 1996

---

***Oak Ridge National Laboratory, managed by Lockheed Martin Energy Research Corp. for the U.S. Department of Energy under contract number DE-AC05-96OR22464.**

# DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

## DISCLAIMER

**Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.**

To appear in "Foundations of Decision/
Information Fusion" Workshop Notes,
Washington, DC, August 7-9, 1996.

# Information Fusion for Automatic Text Classification

Venu Dasigi

Department of Computer Science
and Information Technology
Sacred Heart University
Fairfield, CT 06432-1000
dasigiv@sacredheart.edu

Reinhold C. Mann
Vladimir A. Protopopescu
Computer Science
and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, TN 37831-6364
mannrc@ornl.gov
protopopesva@ornl.gov

## Abstract

Analysis and classification of free text documents encompass decision-making processes that rely on several clues derived from the text and other contextual information. When using multiple clues, it is generally not known a priori how these should be integrated into a decision. An algorithmic sensor based on Latent Semantic Indexing (LSI) - a recent successful method for text retrieval rather than classification [5] - is the primary sensor used in our work, but its utility is limited by the *reference library* of documents. Thus, there is an important need to complement or at least supplement this sensor. We have developed a system that uses a neural network to integrate the LSI-based sensor with other clues derived from the text. This approach allows for systematic fusion of several information sources in order to determine a combined best decision about the category to which a document belongs.

## 1 Introduction

With explosive growth of multi-media data repositories, rapidly increasing connectivity of computer networks and the emergence of and widespread access to the national information infrastructure, there is an urgent need for intelligent tools for access, analysis, and filtering of multi-media information. Significant progress has been made in document retrieval, going beyond the traditional methods of Boolean and exact keyword matching [11] [2] [8] [3] [16]. Much work has been done in applying probabilistic methods to information retrieval (IR) and by representing documents as vectors in an n-dimensional space [14]. Work in natural language-based methods, while quite impressive, often suffers from one major drawback, namely it does not scale up effectively [12]. Integrated approaches that combine natural language methods with statistical and/or vector-based approaches appear to hold promise, but are relatively rare [9].

The general problem is that in analyzing large amounts of text, not any single clue gives enough confidence for proper classification, and when using multiple clues, it is not known how these should be integrated into a decision. Thus, there has been some work in IR on how to best combine information from different algorithms. Multiple algorithms can be combined in at least two possible ways: (i) combining multiple (aspects of) *representation of the same input* into a pattern-matching system (often called *information/sensor fusion*; in the IR literature, this method is often called *query combination*, and (ii) combining *results* from multiple pattern matchers (often called decision fusion; in the IR literature the term *data combination/fusion* is often used) [1] [7] [13]. For this study, we have used neural nets as parameterized mappings that allow for fusion of higher level clues extracted from free text. (This is like query combination, except that we perform classification, rather than retrieval.) An important issue in this context is the containment of the size of the resulting network when multiple input sensors are used.

### 1.1 A Multi-Sensor Neural Net

This work was, in part, motivated by the success of Gene Recognition and Analysis Internet Link (GRAIL), a pattern recognition system which used a multi-layer, feed-forward neural network that receives inputs from several sensors that measure different characteristics of the signals or data sets to be analyzed [17]. The net acts as a classifier and assigns the input pattern to a given number of classes, after being trained. The neural net represents a reliable mechanism to integrate the information from multiple sources to form a combined best estimate of the true classification decision (an example of information fusion). The term "sensor" is interpreted in a broad sense. It can encompass a real physical sensor device, a "logical sensor", (i.e. an algorithm that computes a feature), or a combination thereof [15].

There is much similarity between the problem solved by GRAIL and text classification. Both problems involve decision making based on multiple clues to be derived from large amounts of data and then fusing the information. We use a sensor based on LSI, but its limitations point to the need for other sensors, as explained in the next section. Our hypothesis is that a GRAIL-like system would be very appropriate for classification and filtering of English text documents. We expect the system to be capable of integrating in a systematic way existing and new algorithms as required by the application. The GRAIL-type system can integrate different kinds of sensors, e.g., statistical and syntactic sensors as well as simple keyword sensors, and other standard techniques already in use by document analysis community.

A major stumbling block in applying neural networks to most IR applications has been that the size of a typical IR problem results in impractically large neural networks. An LSI-based approach may be used to achieve the necessary dimensionality reduction and thereby render the approach feasible. In addition to adding trainability, a neural network can integrate the information in inputs coming from other logical sensors into the final outcome.

## 2 Information Sensors

The input to the system is an individual document that needs to be classified into one of several categories. Different logical sensors are applied to the document, constituting different kinds of preprocessing to derive salient features. The primary sensor of interest to this work is based on the term vector representing the input document, which is reduced to a much smaller size using an LSI-based linear transformation. The features derived by the logical sensors constitute input to a neural network that has already been trained. The output is an indication of the category to which the document belongs.

In LSI, a large and sparse term-document matrix is reduced into three relatively small matrices (one of which is simply a diagonal matrix) by singular value decomposition (SVD) corresponding to a number of the dominant singular values. Instead of representing documents by thousands of possible terms, LSI allows a document to be represented by a substantially smaller number of "factors" that are supposed to capture the "significant" term-document associations. This is done by some linear transformations of the much longer term vector, using the constituent matrices that result from the SVD of a *reference matrix* [4].

A reference matrix is the term-document matrix of a *reference library/collection* of documents. A reference library is simply the collection of documents that "adequately" represents all concepts of interest. The idea here that the SVD computations are performed on such a reference library once and for all, and the transformations mentioned above essentially project *any* new document into the "concept space" represented by the reference library.

LSI is a very powerful technique with solid mathematical foundation, and gives rise to the need for a reference library. Indeed it indirectly captures some semantic notions such as polysemy and synonymy for the terms in the reference library [5]. Its utility, however, is limited by the vocabulary seen so far in the reference library. One way to address this limitation is to identify additional sensors that can complement or at least supplement the information provided by the LSI-based sensor. Ideally, such sensors should be sensitive to new words and other patterns in the input or otherwise relate to the output categories.

The purpose of the second logical sensor currently used in this work is to allow for simple keyword profiles to be considered in the classification. Each category profile is simply a set of keywords characterizing that particular category. There is one input to the neural network from this logical sensor, corresponding to each category. Each such input simply represents what fraction of the terms in the given document match the category profile. Inclusion of more sophisticated algorithms is the subject of ongoing research, but as it stands, the information supplied by this second sensor is different from and independent of that from the first LSI-based sensor.

## 3 Experimental Approach

We focused on a number of AP news wire stories from the standard TIPSTER collection [6]. The collection contains AP news wire stories for two full years, tens to hundreds of stories per day. The purpose of the multi-sensor neural net is to classify the news stories into one of ten ad hoc categories, such as accidents, crime, business and finance, culture, politics and government, weather, obituary, etc. For the documents used for training and testing purposes, the categories of the news story documents were manually determined, because category information is not encoded in the TIPSTER collection. This turned out to be a bottleneck.

Despite dimensionality reduction of the input vector through LSI, the neural network is of a substantial size (see below), with more than one hundred input nodes and about ten output nodes. Such a network typically requires several thousand training inputs, and this requirement increases with the number of hidden units. Within the time frame of this initial effort, we manually categorized a few hundred news stories. Consequently, we do not believe the following results represent the best performance of the system. We believe that they do, however, speak for the promise of the approach and allow for comparison of how information from the different sets of sensors can be fused, in spite of the fact that the neural net is inadequately trained.

Our experiments involved a comparison of performance among five approaches, one involving the classic LSI and four based on neural network training. The first approach is the original LSI approach, modified to perform classification by first identifying the document from the reference library that best matches the input document, and then looking up the category of the reference document. Of the neural network methods, two involved just a single sensor and two involved two sensors each. The first single-sensor method (version 1a) used a sensor derived using an LSI-based dimensionality reduction, which was also used as one of the two sensors in both the two-sensor methods. The second single-sensor method (1b) used the simple category profiles, as already explained. The two-sensor methods also used a second sensor based on category profiles. The difference between the two methods was that in one method the category profiles were directly used (2a), and in the other, the results of SVD from the category profiles were used (2b).

The goal in comparing the last two approaches was to see whether the results would be substantially different. The initial expectation was that there should not be any substantial difference, but we wondered if LSI would be able to capture any hidden associations between the category profiles and the documents. This was expected because the case with category profiles is somewhat similar to that of terms in documents. In general, both the term-document matrix and the profile-document matrix have multiple nonzero entries in any given row or column.

Of the nearly five hundred documents used for training, about three-quarters were used as the "reference library" for *all* LSI/SVD operations. The number of SVD "factors" used in this work was 112. We used simple feedforward nets with back propagation, and used the delta rule for learning and the *tanh* transfer function. They were tested in different configurations: two single-sensor versions and two two-sensor versions. One single-sensor version (1a) used the LSI-based term frequency sensor comprising 112 input units and 9 hidden units. The other single-sensor version (1b) used the simple category profile sensor, with 10 input units (one corresponding to each category profile) and 5 hidden units. Both two-sensor versions used the LSI-based term frequency sensor as the first sensor (corresponding to 112 input units), and a second sensor based on the category profiles (corresponding to 10 more input units). The first two-sensor version (2a) used a sensor identical to that of version 1b for its second sensor, and 10 hidden units. The other two-sensor version (2b) used an SVD-based version of the sensor in version 1b as the second sensor, and also used 10 hidden units. All configurations used 10 output units, one for each category.

## 4 Discussion of the Results

When used by itself, the LSI method yielded somewhat surprising results. Indeed, although the performance of LSI in classifying the *known reference library* documents was a perfect 100%, the percentage of correct results when *new documents* outside the reference library were used dropped to 54%. For this method, there was essentially just a single experiment, because there was no training involved and consequently, there was only one non-trivial test set, that of the documents outside the reference library.

For the four neural network experiments, the percentages of correct results as cited represent the peak performance that did not get any better with more iterations. Since there were only a limited number of inputs, several different experiments were constructed by cross validation. We cross validated the available data by generating a dozen pairs of files, each pair containing a training file (90% of data) and a test file (10% of data). The same pairs of data sets were used to test all four neural net configurations, and are referred to by the same data set numbers across the different experiments.

| Single-Sensor Neural Network Classifiers | | | | | | |
|---|---|---|---|---|---|---|
| Data Set No. | Term LSI Sensor | | | Category Profile Sensor | | |
| | No. of Iterations | Correct with Test Data (%) | Correct with Training Data(%) | No. of Iterations | Correct with Test Data (%) | Correct with Training Data(%) |
| 1 | 48K | 58 | 80.93 | 224K | 70 | 64.19 |
| 2 | 48K | 68 | 78.14 | 256K | 66 | 64.42 |
| 3 | 64K | 72 | 77.21 | 128K | 12 | 8.60 |
| 4 | 16K | 62 | 76.74 | 64K | 30 | 24.19 |
| 5 | 96K | 70 | 77.44 | 64K | 34 | 23.72 |
| 6 | 48K | 62 | 78.14 | 64K | 22 | 26.28 |
| 7 | 128K | 62 | 80.23 | 160K | 36 | 28.84 |
| 8 | 64K | 58 | 77.91 | 64K | 14 | 10.00 |
| 9 | 32K | 66 | 77.21 | 64K | 20 | 25.35 |
| 10 | 32K | 60 | 76.98 | 128K | 6 | 7.44 |
| 11 | 48K | 62 | 78.37 | 64K | 22 | 28.60 |
| 12 | 48K | 68 | 76.51 | 160K | 78 | 62.09 |

**Table 1: Results of Classification using neural nets with an LSI-based term frequency sensor input and a simple category profile sensor input**

| Two-Sensor Neural Network Classifiers | | | | | | |
|---|---|---|---|---|---|---|
| Data Set No. | Simple Version of Category Profile Sensor | | | LSI-based Version of Category Profile Sensor | | |
| | No. of Iterations | Correct with Test Data (%) | Correct with Training Data(%) | No. of Iterations | Correct with Test Data (%) | Correct with Training Data(%) |
| 1 | 64K | 58 | 85.12 | 144K | 56 | 84.42 |
| 2 | 32K | 70 | 83.26 | 80K | 70 | 83.26 |
| 3 | 80K | 76 | 84.65 | 80K | 72 | 84.19 |
| 4 | 64K | 66 | 84.19 | 48K | 62 | 83.95 |
| 5 | 80K | 70 | 84.19 | 64K | 62 | 84.42 |
| 6 | 64K | 68 | 82.33 | 64K | 64 | 82.33 |
| 7 | 32K | 68 | 85.12 | 80K | 64 | 83.49 |
| 8 | 32K | 68 | 83.72 | 80K | 62 | 85.81 |
| 9 | 128K | 72 | 85.35 | 80K | 66 | 82.33 |
| 10 | 80K | 64 | 86.05 | 128K | 60 | 83.95 |
| 11 | 32K | 68 | 82.09 | 64K | 68 | 83.02 |
| 12 | 64K | 74 | 84.19 | 48K | 66 | 83.26 |

**Table 2: Results of Classification using two-sensor neural nets, the first with a simple category profile second sensor and the second with an LSI-based category profile sensor**

Tables 1 and 2 summarize the main results of the neural network experiments, which are discussed below.

**Single-sensor neural nets:** Twelve sets of data were used in each experiment. With each training set, to achieve best performance, the neural net was trained for a number of iterations varying between 16,000 and 128,000 for version 1a, and between 64,000 and 256,000 for version 1b. On the test set, the correctness percentage ranged from a minimum of 58% to a maximum of 72% for version 1a and from 6% to 78% for version 1b. When the training set itself was used as a data set, the performance was between 76.05% to 80.7% correct for version 1a and between 7.44% to 64.4version 1b (contrasted to 100% in the LSI-only method). That very large standard deviation is associated with version 1b is not surprising if we take into account the overall performance of this sensor which is rather poor (around 30% on the average, with more iterations in general).

**Two-sensor neural nets:** Again, with each training set, to get the peak performance, the neural net was trained for between 32,000 and 128,000 iterations for version 2a and between 48,000 and 144,000 iterations for version 2b. When the category profiles were directly used for the second sensor (version 2a), the correctness percentage on the test sets ranged from a minimum of 58% to a maximum of 76%. When the training set itself was used as a data set, the performance was between 80.47% to 85.11% correct.

When the results of SVD from the category profiles were used for the second sensor (version 2b), the correctness percentage ranged between 56% and 72% on the test sets and between 82.33% and 85.81% for the training sets. Comparing these results, we notice that the first fusion method (2a) yielded a slight improvement in performance over version 1a (7-8% on the average), while the second fusion method (2b) yielded practically no improvement, although it typically required more iterations than the first fusion method (2a).

The modest improvement in the first case is due to the fact that the second sensor is not very powerful to begin with. We expect that if the two sensors have comparable individual performances and provide relatively uncorrelated information, their fusion should - in general - yield more substantial improvements. However we also expect a certain saturation or even decrease in performance to set in, if both sensors have extreme individual performance.

The lack of improvement in the second case may be related to the number of iterations, the size of the test sets, etc. However, we feel that the main cause for this behavior is the fact that by applying a SVD to the results of the second sensor, one makes it "too parallel" to the first one, thereby stripping an already poor classifier of its discriminating power. A substantially different reason could also contribute to this behavior: the SVD algorithms used in this case might not have been best suited for the case where one dimension of the matrix is only 10.[1]

This work is far from complete. The slight improvement of results even with such a simple addition is encouraging, but also points to the need for more informative/discriminating sensors, better neural network architectures, and more training data. We intend to systematically pursue all these avenues in our future research.

## Acknowledgments

Venu Dasigi thanks the Oak Ridge Institute for Science and Education for the fellowship that made this work possible. The authors thank Professor Mike Berry for distributing the SVDPACKC software.

## References

[1] Belkin, N. J., P. Kantor, C. Cool and R. Quatrain, Combining Evidence for Information Retrieval. In Harman, D. (Ed.), *The Second TREC Conference*, NIST Special Publication 500-215, pp. 35-44, 1994.

[2] Communications of the Association for Computing Machinery, Special Issue on Information Filtering, 35(12), December, 1992.

[3] The Computer Journal, Special Issue on Information Retrieval, the British Computer Society and Cambridge University Press, 35(3), June, 1992.

[4] Dasigi, V. and R. C. Mann, Toward a Multi-Sensor Neural Net Approach to Automatic Text Classification, to appear in *Proc. of IFIP-96, International Federation for Information Processing World Conference on Advanced Information Technology Tools*, Canberra, Australia, September, 1996.

[5] Deerwester, S., S. Dumais, G. Furnas, T. Landauer and R. Harshman, Indexing by Latent Semantic

[1] The size of the term-document matrix for the reference library was 10025x380, but the size of the category profile-document matrix was only 10x380.

Analysis, *Journal of the American Society for Information Science*, 41(6), pp. 391-407, 1990.

[6] Harman, D., Overview of the First TREC Conference, *Proc. of SIGIR-93*, pp. 36-47, 1993.

[7] Dumais, S., Combining Evidence for Effective Information Retrieval, *Working Notes of AAAI Spring Symposium of Machine Learning in Information Access*, pp. 26-30, March, 1996.

[8] IEEE Expert, Special Issue on Using AI in Text-Based Information Retrieval, 8(2), April, 1993.

[9] Jacobs, P., Using Statistical Methods to Improve Knowledge-Based News Categorization, *IEEE Expert*, 8(2), pp. 13-23, April, 1993.

[10] Schütze, H., D. Hull and J, Pedersen, A Comparison of Classifiers and Document Representations for the Routing Problem, *Proc. of SIGIR-95*, pp. 229-237, 1995.

[11] Salton, G., Automatic Text Processing, Addison-Wesley Publishing Company, Reading, MA, 1989.

[12] Smeaton, A., Progress in the Application of Natural Language Processing to Information Retrieval Tasks, *the Computer Journal*, 35(3), pp. 268-278, 1992.

[13] Towell, G., E. M. Voorhees, N. K. Gupta and B. Johnson-Laird, Learning Collection Fusion Strategies for Information Retrieval, *Proc. Twelfth Annual Machine Learning Conference*, Lake Tahoe, July, 1995.

[14] Turtle, H., and B. Croft, A Comparison of Text Retrieval Methods, *the Computer Journal*, 35(3), pp. 279-290, 1992.

[15] Uberbacher, E. C., Y. Xu, R.W. Lee, C.W. Glover, M. Beckerman, R. C. Mann, Image Exploitation Using Multi-Sensor/Neural Network Systems, *Proceedings of the 1995 Applied Imagery and Pattern Recognition Workshop*, Washington DC, October, 1995, SPIE, in press.

[16] van Rijsbergen, C., Probabilistic Retrieval Revisited, *the Computer Journal*, 35(3), pp. 291-298, 1992.

[17] Xu, Y., R. Mural, M. Shah and E. Uberbacher, Recognizing Exons in Genomic Sequence using GRAIL II, *Genetic Engineering, Principles and Methods*, Plenum Press, 15, June, 1994.