# An Application of Expert Network to Clinical Classification and MEDLINE Indexing

Yiming Yang
Christopher G. Chute

Section of Medical Information Resources
Mayo Clinic/Foundation
Rochester, Minnesota 55905 USA

## ABSTRACT

*An effective and efficient learning method, Expert Network (ExpNet), is introduced in this paper. ExpNet predicts the related categories of an arbitrary text based on a search of its nearest neighbors in a set of training texts, and a reasoning from the expert-assigned categories of these neighbors. Evaluations in patient-record text classification and MEDLINE document indexing show a performance of ExpNet in recall and precision comparable to the Linear Least Squares Fit (LLSF) mapping method, and significantly better than other methods tested. We also observed that ExpNet is much more efficient than LLSF in computation. The total training and testing time on the patient-record text collection (6134 texts) was 4 minutes for ExpNet versus 96 minutes for LLSF; on the MEDLINE document collection (2344 documents), the total time was 15 minutes for ExpNet versus 4.6 hours for LLSF. It is evident in this study that human knowledge of text categorization can be statistically learned without expensive computation, and that ExpNet is such a solution.*

## INTRODUCTION

The task of text categorization is to assign predefined categories to a free text according to its contents. Diagnoses in patient records, for example, need to be assigned to insurance categories for the purpose of billing. Citations in a bibliographic database, as another example, need to be indexed using subject categories for the purpose of retrieval. Manual categorization remains the dominant method in practical databases. MEDLINE, for example, spends over two million dollars each year for indexing new entries (about 350,000 per year) by human indexers [1]. There is thus a strong motivation for automatic or semi-automatic text categorization.

A major problem in automatic text categorization is the large vocabulary differences between free texts and canonical categories. That is, a matching method based on shared words ("word-based matching") in a text and a category description would be ineffective, because related concepts are often expressed by different words. Using terminology thesauri ("thesaurus-based matching") attempts to reduce the vocabulary differences. General-purpose thesauri, however, often do not have a sufficient vocabulary coverage crossing different applications [2] [3] [4]. Statistical learning from human decisions in text categorization is another effort [3] [5] [6] [7], and has shown promising results in solving the vocabulary difference problem. Many statistical approaches, however, have a relatively high computation cost. Our Linear Least Squares Fit (LLSF) mapping method, for example, while showing significant improvements over word-based matching and thesaurus-based matching, has a cubic time complexity for its training, which makes it expensive to apply this method to very large data collections. Bayesian belief networks, as another example, have a similar problem in large applications [7].

What we need is a statistical learning method which is highly effective and does not require intensive training. We have found Expert Network to be such a solution.

## METHOD

Expert Network is designed to predict the category or categories of an arbitrary text ("the request") based on previously categorized texts. The basic idea to search "the nearest neighbors" (NNs) of the request in a set of training texts, and to estimate the relevance of a category based on how often this category is assigned to the neighbors. This idea can be traced back to the well known NN classification method which has been studied in pattern recognition for four decades, and used to classify a point in a feature space based on a training sample of previously classified points [8]. The NN approach was later found useful in word pronunciation (to determine the phonemes of a novel word according to the pronunciations of training words) [9] and in text categorization (to categorize Census Bureau documents, for example, according to previously categorized documents) [10] [11]. These applications were generalized in a cognitive paradigm, named Memory-based Reasoning, and characterized by its implementation on

0195-4210/94/$5.00 

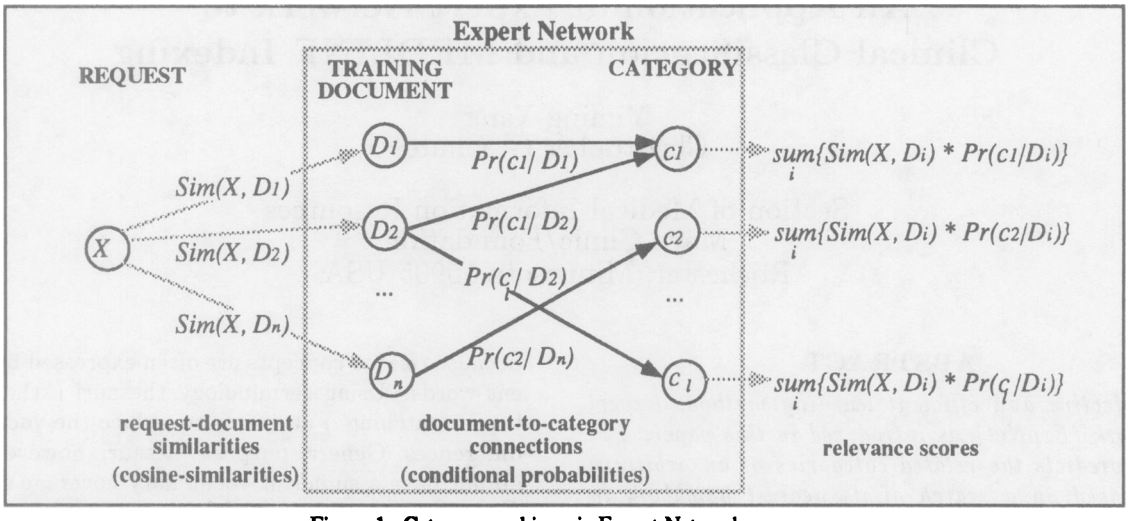


**Figure 1. Category ranking via Expert Network.**

the Connection Machine parallel computers. Our recent study has further pursued the NN approach in both text categorization and retrieval. We use a network formalism to define the method and to encode the statistical evidence of human decisions: a one-layer network is used for text categorization, and a three-layer network is used for text retrieval [12]. Our focus in this paper is on the effectiveness and the efficiency of ExpNet in clinical classification and MEDLINE indexing. In a separate paper, we describe its practical use in assisting human coding of patient record texts at the Section of Medical Information Resources, Mayo Clinic [13].

## The Network

ExpNet is a bipartite network as illustrated in Figure 1. It provides empirical linkages from documents to categories. We use "document" as a generic word for a text which can be the title plus abstract of an article in MEDLINE, or a diagnosis or procedure report in a patient record. A document is treated as a set of weighted words. The input nodes of ExpNet are training documents. The output nodes are the categories of the training documents. The links between documents and categories are weighted using the conditional probabilities of a category being related to a document by human judgment. The conditional probabilities are estimated as the following:

$$P_r(c_k|D_j) \approx \frac{\text{number of times } c_k \text{ is assigned to } D_j}{\text{number of times } D_j \text{ occurs in the sample}}$$

where $D_1, \ldots, D_n$ are unique training documents, and $c_1, \ldots, c_l$ are unique categories. Note that a document may have more than one occurrence in the training sample. Diagnoses in patient records, for example, often repeat. MEDLINE documents, as another example, are unlikely to repeat; however, some may become identical if an aggressive "stoplist" is applied to remove non-informative words. Consequently, the number of times a category is assigned to a document may also be more than one.

## Category Ranking

The category ranking via ExpNet consists of two steps. The first step is to compute the similarity between a given document (the request) and each training document, using the conventional "cosine-measure":

$$sim(X,Y) \stackrel{\text{def}}{=} \frac{\sum_{t_i \in (X \cap Y)} x_i \times y_i}{\sqrt{x_1^2 + x_2^2 + x_3^2 + \ldots} \times \sqrt{y_1^2 + y_2^2 + y_3^2 + \ldots}}$$

where

$X$ and $Y$ are two documents;

$t_i$ is the $i$th word in the document vocabulary;

$x_i$ is the weight of word $t_i$ in $X$;

$y_i$ is the weight of word $t_i$ in $Y$.

For word weighting, we adopted the commonly used schemes as options, including binary weights, within-document term frequency (TF), Inverse Document Frequency (IDF), and the combination TF×IDF [14].

After the similarity values of training documents are computed, these values are propagated to the document-to-category links, multiplied by the weights of these links, and summed at the category nodes. This results in a weighted sum of the conditional probabilities, which we use as the estimated relevance score of a category with respect to the request,

$$rel(c_k|X) \approx \sum_{j=1}^{n} sim(X, D_j) \times P_r(c_k|D_j) \qquad (1)$$

### Optimization

While using a weighted sum of the conditional probabilities is a reasonable way to estimate the relevance of a category, the question is whether we should count all the training documents as the neighbors of an input document. In other words, should we just count the few nearest neighbors and ignore the remaining ones? Would we gain improvement by doing so? To answer these questions, we tested different choices on $n'$ ($n' \leq n$) where $n'$ is the number of selected NNs. Formula (1) is therefore modified as below,

$$rel(c_k|X) \approx \sum_{D_j \in S} sim(X, D_j) \times P_r(c_k|D_j) \quad (2)$$

where $S$ is the set of the $n'$ top-ranking documents.

We used a collection of MEDLINE documents for this test (MEDCL in the next section). We arbitrarily picked a quarter (586 documents) of the total (2344 documents) for training, and used the remaining ones (1758 documents) for testing. There was no overlap between the training documents and the testing documents. The NN selection thresholds were set to $n' = 1, 5, 10, 20, 30, ..., 586$. For each value of $n'$ we computed the precision values at recalls of 10%, 20%, ..., 100% and averaged them for a global measure. Figure 2 shows the result curve. The interesting points are:

(1) the poorest result occurred when $n' = 1$;

(2) the best result occurred when $n' = 30$;

(3) for $n' > 30$, the performance slowly decreased and converged to the level of selecting all the documents ($n'$ =586).

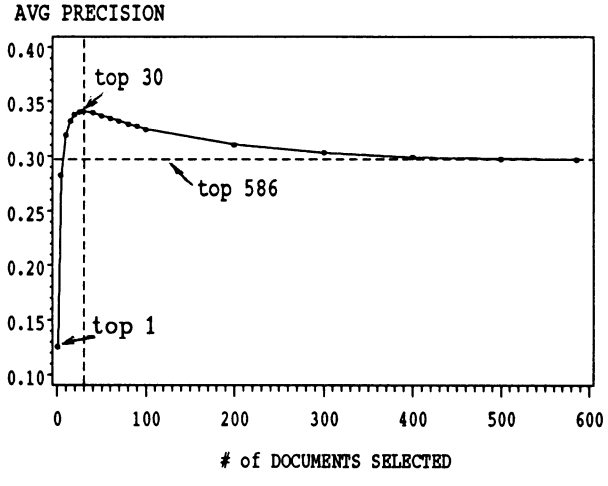


**Figure 2. The effect of document selection**

These testing results suggest:

(1) the top-ranking document by itself would not give sufficient information about the categories of the request, if there is only a partial match between the request and the top-ranking document;

(2) a few top-ranking documents together are much more informative about the contents (categories) of the request;

(3) after a certain point, counting more documents with lower similarity values only contributes noise.

Note that the above observations are based on the test where the training documents and the testing documents are different. This is typically true for bibliographical documents but not necessarily true for patient-record texts because diagnoses or procedure reports are relatively short and often repeat. Our experiences in patient-record text categorization suggest to use the following rules for NN selection:

(1) choose the top-ranking training document if its similarity score is 1 or "sufficiently" close to 1;

(2) choose the $n'$ top-ranking training documents otherwise.

The parameter $n'$ can be empirically determined. For a patient-record text collection (SURCL in the next section), we found that $n'$=10 is the best setting; for the MEDLINE documents, around 20 or 30 are the best choices. The point is, the optimal threshold is application dependent, and the choice should be left to application and experiment.

## EVALUATION

Two text collections were chosen for evaluation, and three different categorization methods were tested for the comparison with ExpNet.

### Data Sets

**SURCL**: a collection of surgical reports from patient records in the Mayo Clinic archive. About 1.5 million patient records are manually coded each year at Mayo for the purpose of billing and research. From the 1990 surgical reports, we arbitrarily chose a cardiovascular subset which contains 6150 procedure/category pairs. We sorted these pairs by category and split them into odd and even halves. The odd-half was used as the training set, and the even-half was used for testing. The average length of texts was about 9 words; 99.8% of them had a uniquely matched category; the rest had two or three categories. There are 281 categories in the cardiovascular subdomain, the procedure volume of the canonical classification system ICD-9-CM (International Classification of Diseases, 9th Revision, Clinical Modifications).

**MEDCL**: a collection of MEDLINE documents. This data set was originally designed for an evaluation of the Boolean search of MEDLINE retrieval [15], and has been used for evaluations of other retrieval and categorization systems [1] [3] [4]. The categories of the documents were assigned by MEDLINE index-

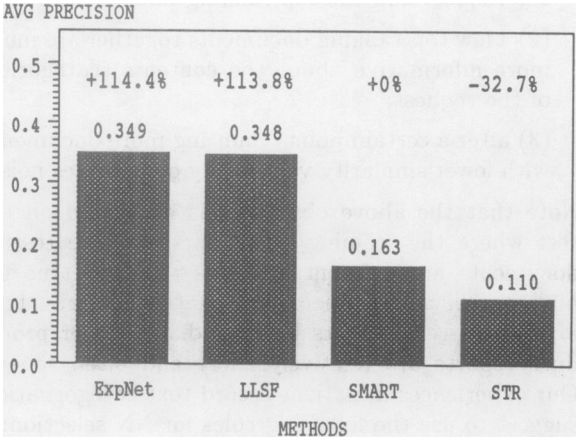


**Figure 3. Different methods on MEDCL**

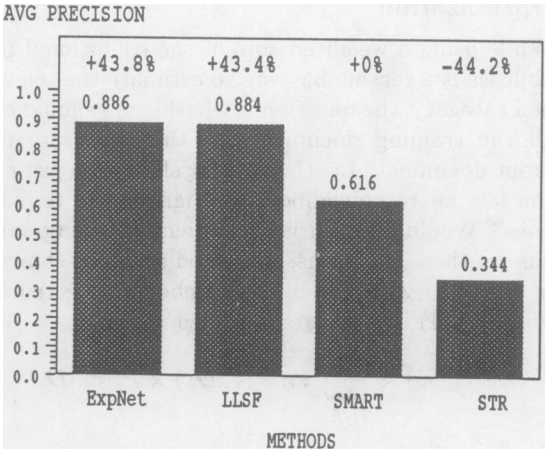


**Figure 4. Different methods on SURCL**

ers; about 17 categories per document on average, and 4020 unique categories in total. The average number of words per document was 168. We arbitrarily used a quarter of the documents for training, and the remaining ones for testing.

A preprocessing was applied to these texts or documents to remove punctuation and numbers, and to change uppercase letters to lowercase; neither stemming nor removal of noise words was applied. The parameter $n'$ in ExpNet was set to 10 for SURCL, and 30 for MEDCL.

### Methods for Comparison with ExpNet

**LLSF**, a statistical learning method which uses the same kind of training data as used in ExpNet, that is, a training sample of manually categorized documents. LLSF computes a mapping function from a document space to weighted combinations of categories. This function guarantees the globally minimized squares error in the mapping.

**STR** (STRing matching) is our implementation of a category ranking method based on shared words. STR represents documents using free words in the documents, and categories using the words in their canonical descriptions. Binary word weights are used for either case. The cosine-similarity between a document and a category is used as the relevance score of the category.

**SMART**, developed by Salton's group [2], is one of the most representative retrieval systems. SMART provides a word-based matching mechanism and allows the use of statistical word weights. We use the SMART software to test the effects of statistical weights on word-based matching. Relevance feedback is not used because it is not applicable [3]. Documents and categories in the tests of SMART are represented in the same way as in STR, except the binary word weights are replaced by the statistical word weights of SMART. The default parameter settings were used in the tests, including word weighting options TF ("nnn" in the SMART nomenclature) and TF×IDF ("atc"). We will refer to the better result (using TF×IDF) among these two choices in the comparison of SMART with other methods. No claim is made that this result is the best possible for SMART.

### Results

Figures 3 and 4 show the testing results on MEDCL and SURCL. All the methods had a better result on SURCL than their performance on MEDCL, indicating that the former was an easier task than the latter. Nevertheless, the relative differences between the methods are more interesting in this comparison. SMART had better performance than STR, indicating the advantage of its statistical word weighting over the binary weighting of STR. ExpNet and LLSF had a similar performance, and both significantly outperformed SMART and STR, showing the benefit of learning human knowledge. By setting SMART as the base of the comparison, the relative improvements of ExpNet and LLSF are between 43.4% and 114.4%.

While ExpNet and LLSF were almost equally effective (under the condition that parameter $n'$ in ExpNet was properly chosen), they differed significantly in computational efficiency. Our current implementation of LLSF uses the LINPACK algorithm for singular value decomposition, which has a time complexity approximately cubic in the number of training documents [6]. In ExpNet, on the other hand, the major computation is to find the nearest neighbors of a request in the training documents. Such a computation can be done in time approximately linear in the number of training documents [12]. The training of LLSF on MEDCL, for exam-

ple, took about 2.25 hours of CPU time on a SUN SPARCstation 10, while the training of ExpNet took only 16 seconds. The testing (categorization) was 5 seconds per document for LLSF and 0.4 seconds per document for ExpNet. Counting the total time including training and testing as a global measure, the computation on MEDCL took 4.6 hours in LLSF but only 15 minutes in ExpNet. On the SURCL set, as another example, the total time was 96 minutes in LLSF and only 4 minutes in ExpNet.

## DISCUSSION

To summarize this paper, a major problem in automatic text categorization is the large vocabulary gap between free text and canonical categories. The vocabulary gap makes the matching methods based on shared words unavoidably ineffective. ExpNet, with its capability of learning from human categorization decisions, solves this problem effectively and efficiently.

The effectiveness and efficiency come from the intelligent use of lexical similarity. ExpNet uses lexical similarity to allocate a request to a neighborhood of training documents where human assigned connections to the related categories are available. The lexical similarity scores of training documents also provide a means to weight and integrate local estimates into a global measure. Since ExpNet does not break training documents into individual words, no assumption of independence among words is made or used in category ranking. Such a use of training documents also keeps the computation of ExpNet much simpler than other statistical learning methods. In LLSF and Bayesian belief networks, for example, the word-category connections have to be made explicitly in their models, which requires intensive computation for the learning.

In conclusion, the simplicity of the model, the effective use of human knowledge, and the efficient computation together make ExpNet a preferable solution to pursue for text categorization. A potential problem we have not focused on in this paper is the real-time response of ExpNet. Since it requires an on-line search of the NNs, the computation must be done in a few seconds. This would be a computational bottleneck when the training sample is very large. Employing parallel computing or distributed computing over multiple computers through networking remains the focus of our future research.

### Acknowledgement

We would like to thank Dr. Kent Bailey for fruitful discussions about the original idea, and Geoffrey Atkin for programming. This work is supported in part by NIH grants LM-07041, LM-05416, and AR30582.

### References

1. Hersh WR, Haynes RB. Evaluation of SAPHIRE: an automated approach to indexing and retrieving medical literature. *Proc 15th Ann Symp Comp Applic Med Care* 1991; 15:808–812

2. Salton G. Development in Automatic Text Retrieval. *Science* 1991; 253:974-980

3. Yang Y, Chute CG. An example-based mapping method for text categorization and retrieval. ACM Transaction on Information Systems 1994; in press

4. Yang Y, Chute CG. Words or Concepts: the Features of Indexing Units and their Optimal Use in Information Retrieval. *Proc 17th Ann Symp Comp Applic Med Care* 1993; 17:685–689

5. Fuhr N, Hartmann S, Lustig G, et al. AIR/X - a rule-based multistage indexing systems for large subject fields. *Proceedings of the RIAO'91* 1991; 606-623

6. Yang Y, Chute CG. A Linear Least Squares Fit mapping method for information retrieval from natural language texts. *Proc 14th International Conference on Computational Linguistics* 1992; 447–453

7. Tzeras K, Hartmann S. Automatic indexing based on Bayesian inference networks. *Proc 16th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval* 1993; 22–34

8. Dasarathy BV, Ed. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques, IEEE Computer Society Press, 1990.

9. Stanfill C, Waltz D. Toward Memory-based Reasoning. *Comm. ACM*, 1986; 29, 1213–1228

10. Creecy RH, Masand BM, Smith SJ, Waltz DL. Trading MIPS and memory for knowledge engineering: classifying census returns on the Connection Machine. *Comm. ACM*, 1992; 35, 48-63.

11. Masand B., Linoff G., Waltz D. Classifying News Stories using Memory Based Reasoning. *15th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval* 1992; 59-64

12. Yang Y. Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval. *17th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994; 13–22

13. Chute CG, Yang Y, Buntrock J. An evaluation of computer-assisted clinical classification algorithms. *18th Ann Symp Comp Applic Med Care* 1994, in press.

14. Salton G. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley, Reading, Pennsylvania, 1989

15. Haynes R, McKibbon K, Walker C, Ryan N, Fitzgerald D, Ramsden M. Online access to MEDLINE in clinical settings. *Ann. Int. Med.* 1990; 112:78–84