Kwan Yi
Library and Information Science
University of Kentucky
502 King library, Lexington, KY 40506
&
Jamshid Beheshti
Graduate School of Library & Information Studies
McGill University
3459 McTavish Street, Montreal, QC H3A 1Y1

Boosting for Text Classification with Subject Headings

Abstracts: The aim of this study is to investigate how Medical Subject Headings (MeSH) as background knowledge source can improve text classification results. The hypothesis is experimented with two different sets of medical documents using HMM-based TC classifier. Experimental results show the improvement of the performance with MeSH in accuracy.

Résumé : Le but de cette étude est d'examiner comment les vedettes-matière médicales (MeSH) en tant que source de connaissances peuvent améliorer les résultats de la classification de textes. L'hypothèse est vérifiée à l'aide de deux différents ensembles de documents médicaux utilisant la classification textuelle basée sur le MCM. Les résultats de cette expérience montrent une amélioration de la performance de précision avec MeSH.

1. Introduction

Text classification (TC) is a relatively new research field spun off from the Information Retrieval (IR) research, with an aim of automatically classifying digital documents to a pre-defined set of classes without human assistance. With the proliferation of digital information available, TC has become more attractive as an automatic information organization tool than ever as a pressing need of acquiring automatic organization tools is increasingly in demand. A primary task of TC is on how a *machine* (generally refer to computer system but use the term *machine* in convention) acquires knowledge needed to determine correct classes of documents. For the last decade, a most dominant approach is based on the framework of machine learning (ML). To some extent the ML approach is similar to human beings in the process of learning knowledge. As humans may gain some knowledge from written materials and documents, a machine also acquires knowledge of a topic or class from documents that were pre-selected for the domain by humans. Such a collection of documents provided is called a *training* set. Therefore, the scope in breadth and depth of the knowledge acquired by a machine is inherently limited by that of a training set. In addition, a machine also needs a systematic scheme for extracting knowledge from a training set, which is in general called a *learning algorithm*, since machine itself performs an automated task only when instructed.

A growing number of researchers from various fields of study, primarily in computer and information science have been interested in the development of automated text-based document classification tools and methods. A broad range of inductive learning algorithms and techniques, such as Support Vector Machines, Baysian Belief Network, Decision Trees, and Artificial Neural Networks, have been proposed and tested, (Yiming

1994; Joachims 1998; Lewis and Ringuette 1994; Mitchell 1997). A learning algorithm and a training set are two primary components affecting TC applications and performances. TC research has heavily focused on the development of effective techniques, methods, and learning algorithms (Sebastiani 2002). In recent years, as TC is more popularly used for real-world problems such as classification of in-patient discharge summaries (Larkey and Croft 1996), flora data classification (Cui, Heidorn, and Zhang 2002), legal document classification (Thompson 2001), and patent document classification (Larkey 1998), researchers are further concerned with the methods and techniques for the improvement of TC performance. An approach for the enhancement may be the use of *prior* background data on classes. A training data set contains some knowledge on target classes but not whole. Therefore, two different TC applications dealing with an identical set of classes acquire different knowledge (so-called *posterior* knowledge) with a different training data set. While many TC systems have been developed in the past, few use *prior* background data on classes, due to either the scarcity of such data, the limited use, or out of major foci.

Medical Subject Headings (MeSH) is known as a popularly-used controlled vocabulary for the health sciences, and provides comprehensive medical subject information with the hierarchical structure of subject terms (Chan 1994). In this study, we propose MeSH as a reliable source of the *prior* information for classes for the following reasons: 1) MeSH is a specialized *subject* vocabulary set; 2) MeSH is a controlled vocabulary *thesaurus* – a comprehensive list of all valid subject terms with semantic relationship among associated terms; 3) All the subject terms in MeSH are arranged under a single hierarchical structure. The hypothesis of this study is that additional *prior* background data generally covering target classes to training data (*posterior* knowledge) boost the performance of TC systems. The objective of this study is to analyze the effect of a medical knowledge source in creating classifiers for medical documents. We seek answer to the following question with empirical evidence: To what extend can the knowledge of MeSH as a source of the *prior* information for target classes improve the performance of a TC system?

Section 2 describes the related works. Section 3 describes the classifier and the experimental settings. Section 4 presents the experiments and the results. Section 5 summarizes the conclusion.

2. Background and Related Works

2.1 Understanding of TC

A TC may be seen as the task of assigning a pre-defined set of classes to documents. Its three components may be expressed in an operational definition: analyzing textual documents, understanding their relevant classes, and classifying them into a pre-defined set of classes. Some notations related to TC tasks are defined, which will be used throughout this dissertation. Let $D = \{d_1, d_2, \cdots, d_n\}$ be a set of documents, and $C = \{c_1, c_2, \cdots, c_m\}$ be a set of classes. Also, define C^* to be the power set of C, that is, the set of all subsets of C. More formally, a TC can be viewed as a function $F : d \xrightarrow{c} \{0,1\}$, where $d \in D$ is a document and $c \in C$ is a class, that accepts a document as an input and yields true as 1 or false 0, given a class. The output of 1 means that the document is interpreted to fall into the considering class, and it is interpreted not to fall into the class with the output of 0. The primary element of a TC system is the realization of a function F. As described in the proposed TC definition, the function serves to measure the

relevance of a document given a class. The scope and methods where the function works can be limited by the TC conceptual model on which it relies. TC tasks can be classified into different types, according to the number of classes and the number of class labels. If there are only two classes to be considered, it is said to be a binary classification task, where the value of m in the set C is equal to 2. With more than two classes, it is called a multi-class classification, where the value of m is larger than 2. When each document is associated with one class label, it is called a binary-label classification, where the class label of a document is an element of the set C. In multi-label classification, each document has at least one class label, where the class label of a document is an element of the set C^*.

2.1 Machine Learning Approach to TC

A general machine learning problem is identified as a process of three components of (1) task, (2) performance measure, and (3) experience (Mitchell 1997). The notion of *learning* from the ML perspective has the same base as the one for human learning, but focuses on the machine's performance. Learning in ML is viewed as the process of improving the system's performance by acquiring knowledge from experience (Langley 1996). A ML framework more closely associated with TC can be found in Kubat et al. (1999). In the framework, the four components - *examples*, *background knowledge*, *learning algorithm*, and *concept description* - are co-related as input, output, and a black box (function) in between. *Examples* (also called samples, instances, input vectors or feature vectors) as an input to the black box convey knowledge for the target concept to be learned. An *example* consists of a *description* of the concept and *value* to the description. The *description* of an *example* is represented in various forms, such as vector, rule, and list of terms. The decision of the form for the description is subject to the learning algorithm applied in the black box. One of the most popular forms is the vector-based representation. In this case, each component of a vector is called a feature (also, called attribute or variable). The *value* of an *example* can have any value among a finite set of numbers or categorical values, subject to the target concept. In most classification tasks of ML, the *value* is binary (for example, true or false), which specifies a positive or negative *example*. Let X be an *example*. A vector-based representation is X: $(X_1, \ldots, X_n)$ = true. *Background knowledge* as another input to a learning algorithm in the framework consists of prior knowledge about the target concept to be learned. For example, when a chess game is considered to be a learning task, chess rules could be the background knowledge in the ML framework, whereas a set of the changed chessboard positions can be the examples for the experience of the chess task. Inductive ML methods, such as decision tree learning and neural networks, are inductive learning paradigms that make generalizations about a target concept based mainly on a number of training examples. Prior knowledge is the main resource component for analytic methods, along with training examples. *Learning algorithm* serving as a black box in the ML framework is the learning method that is concerned with how to learn and what is to be learned. *Concept description* is the realization of what a ML model learns from the set of examples used as the target concept. Thus, it can be different due to the type of knowledge representation and training examples collected. A general goal of ML is to make the machine gain knowledge from previous experience. In the inductive ML approach, the target knowledge will be achieved in a representation form specific to the learning algorithm, and a learning algorithm is applied in such a way that the parameters of the knowledge representative structure are trained. Different ML paradigms support different representations of knowledge, and adopt different learning methods. In neural network algorithms, knowledge is represented as a graph consisting of nodes and edges, and, in

rule induction, condition-action rules are used. In other methods, functions, logic programs and rule sets, finite-state machines, grammars, and problem solving systems have been adopted to represent knowledge.

2.2 Use of Background Knowledge Sources in TC

Various knowledge sources and methods were used to improve text classifiers.

Metadata classification
One of the pioneer works on automated classification can date back to Larson's work (1992), where he attempted to classify a set of Machine Readable-Cataloging (MARC) records into Library of Congress Classification (LCC), based on title and subject headings appearing on MARC records. It was concluded that the use of the first subject heading only was the most effective in the automatic task, as opposed to various combinations of the title and subject headings. A similar work can be found in the Pharos project (Dolin, Agrawal, and El Abbadi 1999) as a part of the Alexandria Digital Library project. In the project, 1.5 million cataloging records from the University of California Santa Barbara library were classified based on the Latent Semantic Indexing method. However, a comprehensive evaluation for the classification system is not reported yet. The most recent work directly linked to Larson's work can be found in the INFOMINE project (Frank and Paynter 2004). The project aims to predict and assign a Library of Congress Classification (LCC) to library cataloging records of Internet resources using Library of Congress Subject Headings (LCSH). The classification accuracy of the classifier is reported to be the range from 55% to 80% approximately, depending on the window of top ranking.

Web classification
The DESIRE project (Koch and Day 1997) is a large-scale international project with an aim of developing a high quality research information database for research community in the European Union. The project includes the automatic classification of assigning the Engineering Information (EI) categories to Web documents and electronic resources. The metadata, headings, and plain text of the Web documents are matched against the terms from the EI thesaurus representing an EI classification category. Furthermore, LCSH were added to the terms representing classification categories, and the words and phrases from documents, rather than the full-text, were intended to be matched.

Document classification
The Unified Medical Language System (UMLS) Metathesaurus, developed and distributed by the National Library of Medicine, contains biomedical terms from a number of controlled vocabularies. Wilcox et al. (2000) investigated the use of a natural language processor and controlled vocabulary metathesaurus to improve the performance of text classifiers for medical documents. They demonstrated that the incorporation of two knowledge sources adopted improve the classifier's performance. In (Zelikovitz and Hirsh 2003), various external knowledge sources to the primary source to be classified were utilized. In the classification of journal titles, abstracts and reviews were used by classifiers. Methods of incorporating background knowledge into text classification were presented.

3. A Text Classification System

The HMM theory has been applied in some TC-related applications (Conroy and O'Leary 2001; Miller, Leek, and Schwartz 1999; Freitag and McCallum 2000). In this experiment, a modified version of the HMM-based TCs (Yi 2005) is used as a base system.

3.1 A HMM-based Classification Model

A Markov model is a statistical model that derives from a Markov theory in early 1900s, and it may be viewed as a transitional diagram composed of states and transitions between states. A hidden Markov model is the extension of a (observable) Markov model, and differs from an observable model in that a sequence of states corresponding to a list of observations is not immediately observable. In an observable Markov model, an observation is corresponding to a state of the model, whereas, in a hidden Markov model, it is a probability function of a state, not a state itself. Therefore, given an observation sequence, the corresponding path (a sequence of states) is uniquely obtained with a Markov model, whereas a unique corresponding path is not recovered with a HMM. Regardless of observable or not, a Markov model is based on an assumption that the future is predicted only with the data at the present, rather than considering all the past, which is called *Markov condition*. The application of the Markov condition to Markov models is described in the following. For the probability of a sequence of random variables, $P(X^{(0)},...,X^{(m+1)})$ to be calculated, all the previous random variables need to be involved for the calculation, as shown below:

$$P(X^{(0)},...,X^{(m+1)}) = P(X^{(0)})P(X^{(1)} \mid X^{(0)}) \cdots P(X^{(m)} \mid X^{(0)},...,X^{(m-1)})P(X^{(m+1)} \mid X^{(0)},...,X^{(m)}).$$

However, the assumption of the *Markov condition* replaces all past history of a random variable only with a current random variable, which is shown below:

$$P(X^{(0)},...,X^{(m+1)}) = P(X^{(0)})P(X^{(1)} \mid X^{(0)}) \cdots P(X^{(m)} \mid X^{(m-1)})P(X^{(m+1)} \mid X^{(m)}).$$

Despite the criticism to the simplified assumption of being unrealistic and non-practical, it has been well performed in a wide range of applications, especially signal processing and speech recognition (Rabiner 1989).

A HMM M is represented by the following characteristic components: $M = (I, E, T, O, S)$ where I being initial probabilities, E output symbol emission probabilities, T state transition probabilities, O a set of output symbols, and S a set of states. The proposed HMM system has been designed to reflect the idea of making a different information source as a separate state and of all being statistically combined. Thus, a set of different states is corresponding to a number of different sources. The set of output symbols O used in this model is viewed as terms from all ISs adopted in the model. Two assumptions governing HMM are summarized by: (1) an emission probability of a symbol is only subject to the current state; (2) the current state is dependent on only one previous state, instead of the history of all previous states, by the rule of Markov condition.

The architecture of the proposed HMM-based TC model is illustrated in Figure 1, which is applied to all TC models for different categories. As shown, except for two dummy states for *start* and *end*, the model has three internal states appearing inside. An internal state specifies a corresponding information source. In the current model, three different information sources (ISs) are implemented, with the possible extension of adding other ISs in future. The first IS labeled *Abstract,* denotes a collection of document abstracts, the second IS labeled *Title* denotes a collection of document titles, and the third IS labeled *MeSH* denotes terms from 2005 MeSH. A set of medical documents is collected from MEDLINE database and used for training this system, and the corresponding *titles* and *abstracts* are employed as sources for the first two ISs, respectively. MeSH is a second source of information for the classification. MeSH is a hierarchical tree structure of fifteen categories at the top level. The electronic-version of 2005 MeSH terms is available at 2005 ASCII MeSH from the National Library of Medicine web site.

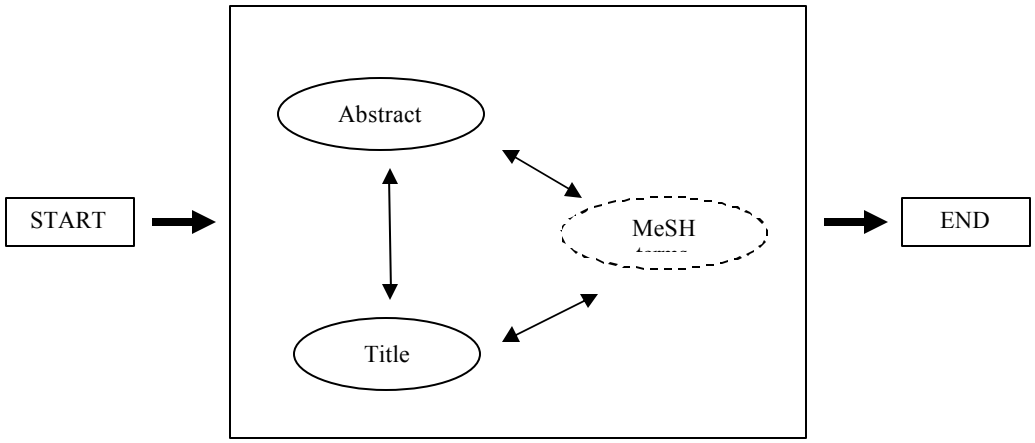

Figure 1: An Architecture of a HMM-based TC System

3.2 Parameter Estimation

Once an architecture of HMM structure is determined including the number of states, their roles, and specification of probabilities, the next procedure is to parameterize the model variables for emission probabilities and state transition probabilities .

As for training, 100,000 medical documents from MEDLINE databases are used to build a simple HMM TC system and MeSH terms are added for a other HMM TC for twenty three categories under the 'diseases' MeSH category (a more detailed description is provided in the *Text Collections* section later). Since all training data collected are pre-classified and labeled with corresponding classes, the emission probabilities of output symbols in a state can be estimated by the ratio of the number of occurrences of a symbol given the total number of all the output symbols (as shown in (1)) that indicates emission probability of the symbol W_i at the state S_j where V is a set of distinct symbols in the state. The n-estimate probability (Mitchell 1997) is adopted to provide a constant probability by a ratio of the number of total symbols for new symbols, in case of generating a zero probability when a symbol does not appear in training data.,

$$P(W_i / S_j, C_c) = \frac{1 + N(W_i, S_j, C_c)}{|V| + \sum_{k=1}^{|V|} N(W_k, S_j, C_c)} \qquad (1)$$

where $N(W_i,S_j,C_c)$ is the total number of word occurrences W_i in training documents whose target class is c and target information source is j. $|V|$ is the total number of distinct terms appearing in the training documents for class c and information source j.

Our model supports the notion of a transition probability going to a state with an average amount of information per symbol (term) in that state. Transition probabilities shifting from the *start* state to another are interpreted as the normalized probabilities of the amount of information calculated for each of internal states. The quantity of information in a state is measured based on Salson's standard term frequency (TF) / inverse document frequency (IDF) (Salton and Buckley 1988) as follows:

$$I(C_i) = \sum_{w \in Ci} I(w) = \sum_{w \in Ci} TF(w, C_i) IDF(w)$$
$$I_{normal}(C_i) = I(C_i) / |C_i|$$
$$P(C_i) = \frac{I_{normal}(C_i)}{\sum_{\forall j} I_{normal}(C_j)} \qquad (2)$$

The states transition probability is the same as the initial probability except of category being considered in addition to class. In summary, an initial probability to a state is estimated by the normalized information quantity belonging to the state, whereas a state transition probability is by the normalized information quantity among states given a same class. For the choice of the TF/IDF algorithm, the version of TF/IDF algorithm used in the Okapi IR system (Robertson et al. 1995) is employed, due to its wide adaptation in IR systems (Ponte and Croft 1998).

3.3 Prediction of Class

The goal of the present work is to classify documents into twenty-three sub-classes of the 'disease' top-level MeSH class. The TC system produces a ranked list of the twenty-three classes as an output for a tested document in the following way. First, we compute a score for every possible path starting from the *start* state and ending with getting to the *end* state. Second, a highest score is picked for the score for a tested document given a class. Third, for a same document, the class matched with the highest score is regarded as the document class.

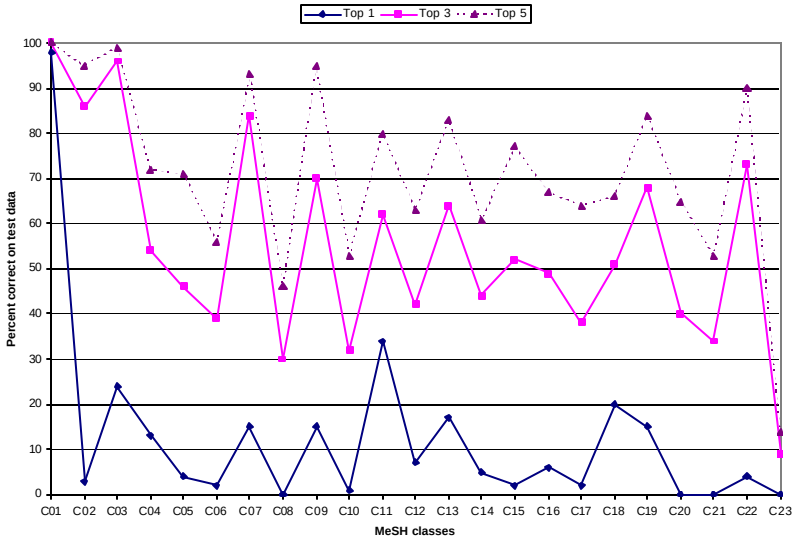

Figure 2: Accuracy on the OHSUMED test data (2261) to the system trained upon OHSUMED.

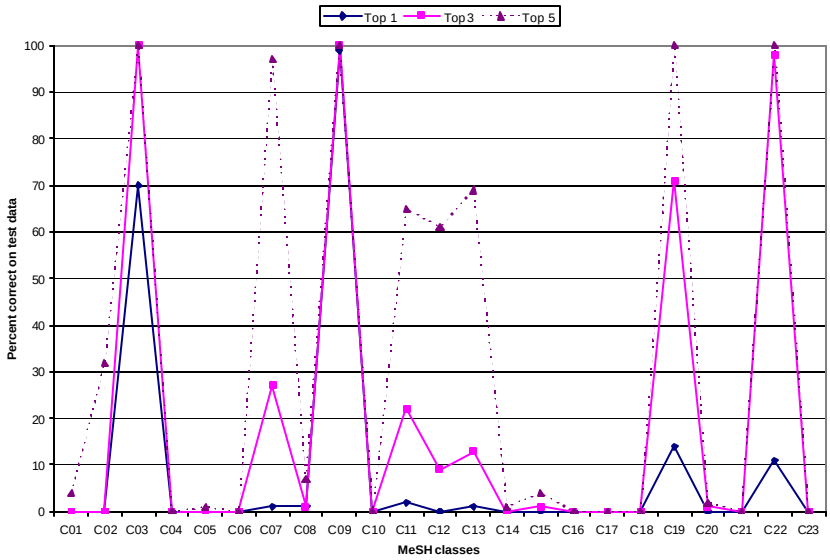

Figure 3: Accuracy on the OHSUMED test data (2261) to the system trained upon OHSUMED and MeSH combined.

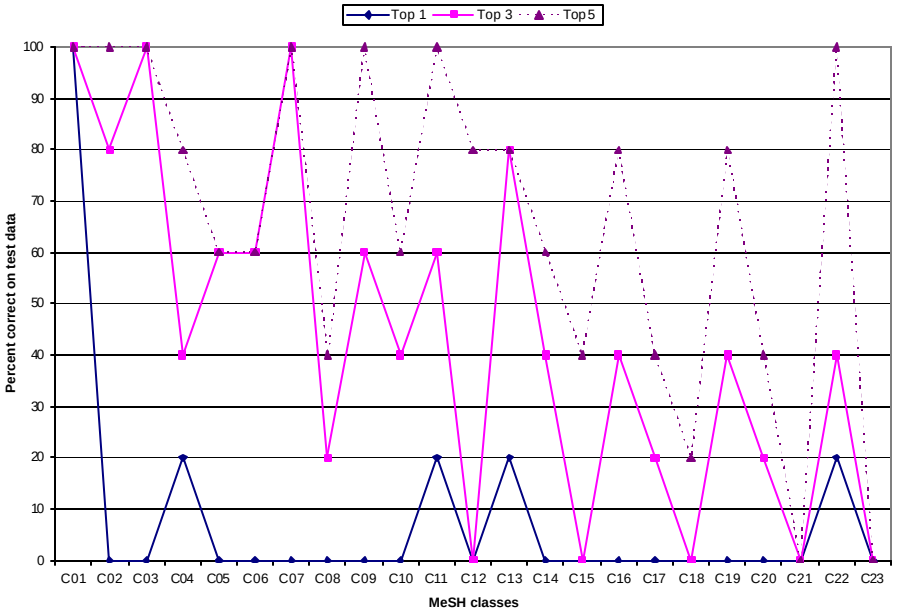

Figure 4: Accuracy on the in-house test data (115) to the system trained upon OHSUMED.

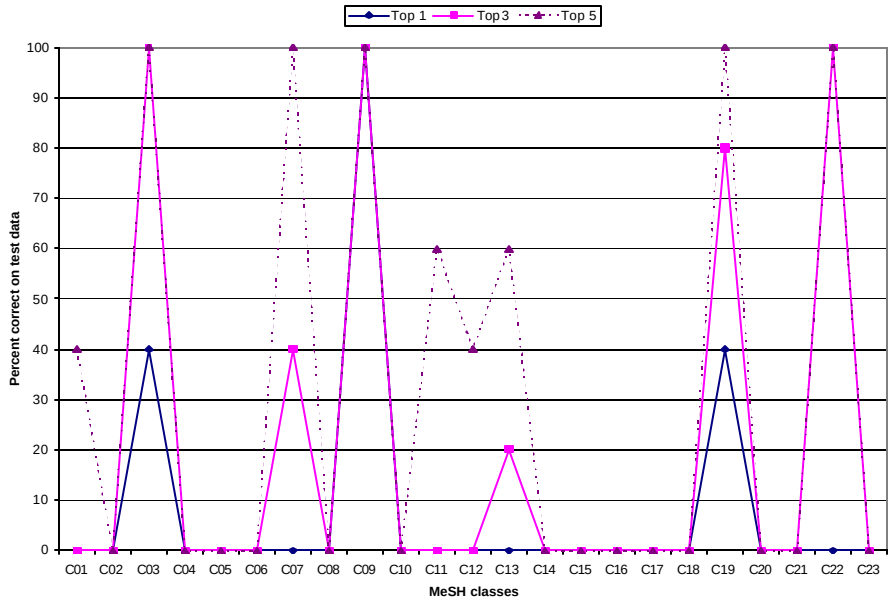

Figure 5: Accuracy on the in-house test data (115) to the system trained upon OHSUMED and MeSH combined.

		Training Data	
		OHSUMED (10,000)	OHSUMED (10,000) + MeSH
Test Data	OHSUMED (2,261)	4.9	9.8
	Other medical documents (115)	5.4	10.0

Table 1: Summary of four different experiments with the averaged accuracy of correct percent.

4. Empirical Results

4.1 Test Collections

The empirical evaluation has been done on two test collections. The first test collection is taken from the OHSUMED corpus (ftp://medir.ohsu.edu/pub/ohsumed/) consisting of 348,566 medical documents excerpted from 270 medical journals published from 1987 to 1991 through the MEDLINE database, which is originally compiled by William Hersh et al. (1994). For each of the MEDLINE document, title, publication type, abstract, author, and source information are provided. Out of 50,216 original documents for the year 1991, the first 20,000 documents are classified into the 23 MeSH 'disease' categories and labelled with one or multiple categories by T. Joachims (1998). The first 10,000 is used for training and the remaining 10,000 is for testing, respectively, in his TC experiment. The same training documents are used to train the classifier built for this experiment. As for test, the first 100 documents for each category are selected from the 10,000 test data. The total number of test documents is 2261[1] in total.

The second test collection is also collected from the MEDLINE database. All data are collected by conducting MEDLINE searches based on the terms found in the twenty-three categories under the main MeSH tree of Diseases. The document search was limited to the terms found in Major Mesh Descriptors (MJME). The collected data was further limited to the first five titles and abstracts of articles returned with the desired MeSH classifications, published in 2004. Data collected was also restricted to documents in the English language. Once the appropriate data was identified, the title and abstract of the article was saved and assigned a file name according to the MeSH classification (i.e. a title and abstract classified by MEDLINE under cardiovascular disease was assigned a file name containing C14). Five titles and abstracts, representative of each Disease subcategory, were ultimately collected and compiled for a total of 115 documents.

4.2 Performance Measure

Classification accuracy is a method used for the estimation of model effectiveness in terms of the correctness of classification. It is measured by the probability of the number of correct classifications over the total number of classifications. Classification accuracy of a model M on a test data set T is measured with the formula below (Mladenic 1998):

[1] There are 70 and 91 documents in C03 and C22 classes, respectively.

$$Accuracy(M,T) = \sum_{e \in T} P(e) \times Correct(e)$$
$$Correct(e) = \begin{Bmatrix} 1 \ if \ C(e) = \hat{C}(e) \\ 0 \quad otherwise \end{Bmatrix} \qquad (3)$$

where $P(e)$ is a probability of a document (example or instance) e that is usually equally considered for all documents, that is, 1/N with N = |T| being the number of documents in the test data set T, $C(e)$ is the *actual* class of a document e, and $C(e)$ refers to the *predicted* class by the model M. It is assumed that $P(e)$ is a constant to all documents. If the model M perfectly estimates the actual classes of all the test documents, then the value of *Accuracy*(M,T) in (3) becomes 1, $\sum_N \left(\frac{1}{N} \times 1 \right) = 1$. On the contrary, if the model does not correctly estimate the actual class of a single document at all, the classification accuracy becomes 0, $\sum_N \left(\frac{1}{N} \times 0 \right) = 0$. As a result, a classification accuracy value based on (3) is between 0 for the worst expectation and 1 for the perfect expectation, and its probabilistic conversion (0% for 0 and 100% for 1) is used in the evaluation at the next section. Since it is defined for binary classification tasks, the accuracy is treated within top-n for the multiple class tasks of this experiment.

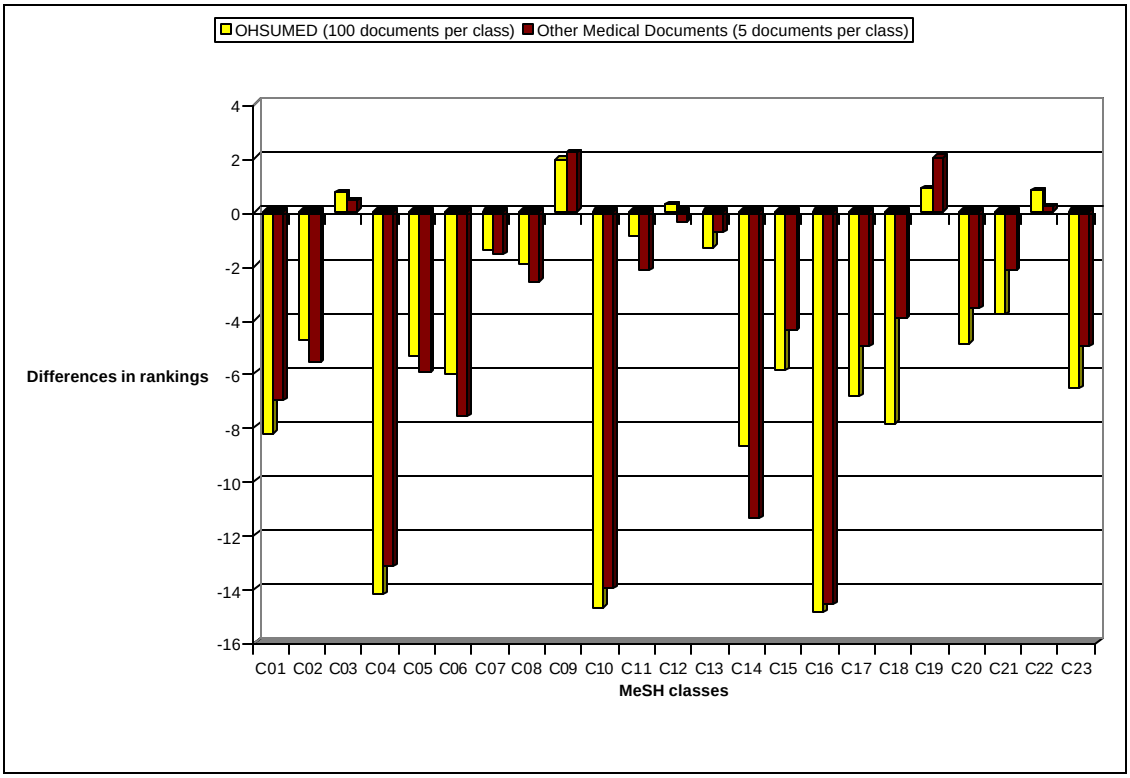

Figure 6: Performance of the classifier trained on the combined OSHUMED and MeSH compared to the one trained on OHSUMED only, in terms of averaged ranking.

4.3 Evaluation

Four different experimental milieus are set by the combination of two different sets of test and training data, respectively. The two training data sets are: (1) 10,000 OSHMED

documents and (2) 10,000 OSHMED documents plus MeSH terms; and the test data sets used are: (1) 2,261 OSUMED documents and (2) 115 medical documents. The result for each case is plotted in Figure 1 through 4. For each case, graphs for three different levels of accuracy are plotted, corresponding to top-1, top-3, and top-5. In an accuracy graph for top-n, a test document is regarded as being correctly classified when an actual class of the document is located within the top n on its ranked list.

On the 2,261 OSUMED test data the performance for the C01 class is extremely high even at the top-1 (see Figure 1) and it dropped down to zero percent at the other case (see Figure 2). However, the opposite situation occurs at the C03 class. A pattern found under the two settings, more clearly depicted with top-3 and top-5, is that the accuracy in Figure 1 is rather stable but that in Figure 2 is more fluctuant across classes. An extreme case is found at the C09 class in Figure 2, where it goes up to the perfect score in top-1.As shown in Table 1, in overall, the classifier trained on OSUMED only performs better than the one trained on the combination of OSUMED and MeSH data. However, more cases highly closed to the perfect accuracy are found in Figure 2 than in Figure 1. On the 115 other medical test data the situation is similar to with the previous comparison.

The empirical data are analyzed based on ranking. Two classes are associated with a test document. One is actual class pre-labeled and the other is the class labeled by the TC system (called it machine-generated class). In measuring accuracy, a machine-generated class within top-n is regarded as correct. If it is out of the range, it is not counted and disregarded. In the view of ranking, the rankings of all machine-generated classes are summarized and averaged, regardless of the range of value. Figure 5 illustrated a relative score by subtracting the averaged ranking with OHSUMED by that with OHSUMED and MeSH combined. The final scores depicted in the Figure can be an indicator how much the combined case contributes to the performance with the OHSUMED case as a basis, in terms of ranking. As illustrated in the Figure, only in a few classes, the combined case influences in positive direction, but there are negative effects in many other classes.

5. Conclusion and Future Work

We have presented a HMM-based classification for a potential role of a subject heading as a source of background information. The argument is experimented with MeSH and assessed in two aspects: accuracy and averaged ranking. The experimental results are summarized in Table 1 and Figure 6 for two perspectives, respectively. As a conclusion, regardless of the test data set used, this pilot study is not successful to discover empirical evidences to support the alleged role of a subject heading. However, it does not fail to provide a lesson that a further sophisticated process is required in manipulating MeSH.

In future work we will evaluate the methods on the various numbers of training data, where we expect to see if the performance stays constant. In addition, more richer data sets will be used to test, by which we see if it confirms the pilot study. More importantly, we will take the structure of MeSH into consideration and link it to differentiate MeSH terms, to analyze the MeSH term more precisely. We will also compare with LCSH. In relation to these methods, we expect our methods to produce more strong evidence toward the potential role.

6. Acknowledgments

The authors would like to thank the School of Library and Information Science at University of Kentucky for the financial support to data collection.

References

Chan, Lois Mai. 1994. *Cataloging and Classification: an Introduction*. New York: McGraw-Hill.

Conroy, J. M., and D. P. O'Leary. 2001. Text summarization via hidden Markov models. *SIGIR Forum*:406-407.

Cui, Hong, P. B. Heidorn, and Hong Zhang. 2002. An approach to automatic classification of text for information retrieval. Paper read at The 2nd ACM/IEEE-CS Joint Conference on Digital Libraries, at Portland, Oregon.

Dolin, R., D. Agrawal, and A. El Abbadi. 1999. Scalable collection summarization and selection, at Berkley, CA, USA.

Frank, E., and G. W. Paynter. 2004. Predicting Library of Congress classifications from Library of Congress subject headings. *Journal of the American Society for Information Science and Technology* 55 (3):214-227.

Freitag, Dayne, and Andrew McCallum. 2000. Information Extraction with HMM Structures Learned by Stochastic Optimization. Paper read at The 17th National Conference on Artificial Intelligence, July 30 - August 3, 2000, at Austin, TX.

Hersh, William, Chris Buckley, T. J. Leone, and David Hickam. 1994. OHSUMED: an interactive retrieval evaluation and new large test collection for research. Paper read at The 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 03-06, 1994, at Dublin, Ireland

Joachims, T. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. Paper read at The 10th European Conference on Machine Learning, April 21-24, 1998, at Chemnitz, Germany.

Koch, Traugott, and Michael Day. *The role of classification schemes in Internet resource description and discovery* 1997 [cited. Available from http://www.ub2.lu.se/desire/radar/reports/D3.2.3/

Kubat, Miroslav, Ivan Bratko, and Ryszard S. Michalski. 1999. A Review of Machine Learning Methods. In *Machine Learning and Data Mining: Methods and Applications*, edited by R. S. Michalski, I. Bratko and M. Kubat. Chichester, England: John Wiley & Sons.

Langley, Pat. 1996. *Elements of Machine Learning*. San Francisco: Morgan Kaufmann.

Larkey, L. S. 1998. Automatic essay grading using text categorization techniques. Paper read at The 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, at Melbourne, Australia.

Larkey, L. S. , and W. B. Croft. 1996. Combining classifiers in text categorization. Paper read at The 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 18-22, 1996, at Zurich, Switzerland.

Larson, R. R. 1992. Experiments in automatic Library of Congress Classification. *Journal of the American Society for Information Science* 43 (2):130-148.

Lewis, D. D., and M. Ringuette. 1994. A Comparison of Two Learning Algorithms for Text Categorization. Paper read at The 3rd Annual Symposium on Document Analysis and Information Retrieval, at Las Vegas, NV.

Miller, D. R. H., T. Leek, and R. M. Schwartz. 1999. A hidden Markov model information retrieval system, at Berkeley, CA, USA.

Mitchell, Tom M. 1997. *Machine Learning*. Boston, MA: McGraw-Hill.
Mladenic, Dunja. 1998. Machine learning on non-homogeneous, distributed text data, Computer Science, University of Ljubljana, Slovenia.
Ponte, J. M., and W. B. Croft. 1998. A language modeling approach to information retrieval, at Melbourne, Vic., Australia.
Rabiner, L. R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77 (2):257-286.
Robertson, S. E., S. Jones S. Walker, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at TREC-3. Paper read at The Third Text Retrieval Conference, TREC-3 NIST Special Publications.
Salton, G., and C. Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management* 24 (5):513-523.
Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys* 34 (1):1-47.
Thompson, Paul. 2001. Automatic categorization of case law. Paper read at The 8th International Conference on Artificial Intelligence and Law, May 2001, at St. Louis, Missouri.
Wilcox, Adam B. , George Hripcsak, and C. Friedman. 2000. The Role of Domain Knowledge in Automating Medical Text Report Classification. In *Workshop on Text Mining in The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* Boston, MA.
Yi, Kwan. 2005. Text Classification Using a Hidden Markov Model, Graduate School of Library and Information Studies, McGill University, Montreal.
Yiming, Yang. 1994. Expert Network: effective and efficient learning from human decisions in text categorization and retrieval, at Dublin, Ireland.
Zelikovitz, Sarah , and Haym Hirsh. 2003. Integrating Background Knowledge with Text Classifiers Paper read at The International Joint Conference of Artificial Intelligence, at Acapulco, Mexico.