

©BRAND X, PHOTODISC

Multilabel Associative Classification Categorization of MEDLINE Articles into MeSH Keywords

An Intelligent Data Mining Technique to More Accurately Classify Large Volumes of Documents

BY RAFAL RAK, LUKASZ A. KURGAN,
AND MAREK REFORMAT

The constantly increasing numbers of scientific documents and the extensive manual work associated with their description and classification requires intelligent classification capabilities for users to find required information. The article discusses an automated method for classification of medical articles into the structure of document repositories, which would support currently performed extensive manual work.

Understanding the Problem

Exponential growth of the number of scientific documents results in increased difficulty in their categorization. This motivates our research in providing intelligent categorization methods. MEDLINE is the National Library of Medicine's (NLM) database consisting of approximately 13 million article references to biomedical journal articles dating back to 1966. NLM employees add approximately 1,500 to 3,500 new article references every day to this database. This includes manual assignment of each article to the corresponding entries in the medical subject headings (MeSH). MeSH is NLM's controlled vocabulary thesaurus consisting of medical terms at various levels of specificity. The rapidly growing number of incoming documents together with error-prone manual work may make this task difficult.

This article describes development of a novel system for automated classification of MEDLINE article references. We employ and redesign a recently developed data mining-based classification tool, namely the associative classifier with reoccurring items (ACRI), to assign MeSH keywords to article references. The method is capable of performing challenging multilabel classification, where the goal is to assign many classes to an object. It is designed and tested on the OHSUMED corpus [1], which consists of a comprehensive set of almost 350,000 article references.

Our goal is to build a multilabel classification system, which specifically aims at applications to the MEDLINE data, based on associative classification. Although a multiclass classification problem, where one out of many classes is assigned to an object, has been widely studied, a relatively small amount of research has been dedicated to investigation of multilabel classification, especially using associative classification. We compare several different associative- classification-

based approaches capable of accomplishing this task. The methods are extensively tested by employing five different approaches to multilabel associative classification to choose an optimal configuration. We also investigate several different measures of classification quality that result in alternative setups and different performance characteristics.

This contribution is an extended version of our article published in the proceeding of the ICMLA [2].

Related Work

NLM's tools and databases have attracted significant attention in recent years. The Text Analysis and Knowledge Mining for Biomedical Documents (MedTAKIMI), which is an application to facilitate knowledge discovery from very large text databases such as the MEDLINE, has been developed and described in [3]. According to the authors, the application dynamically mines documents to obtain their characteristic features and uses categories such as MeSH keywords for term extraction and interactive series of drill-down queries. Another application, called MedMeSH Summarizer, uses MeSH keywords to annotate a set of genes obtained from DNA microarrays by summarizing all the terms tagged to MEDLINE article references that are related to a gene in a user-defined query [4]. Exploration of relationships between features used to represent text, with application to MEDLINE, has been studied in [5]. The method uses association rules and compares three different semantic levels: words, MeSH keywords, and automatically selected concepts coming from NLM's Unified Medical Language System (UMLS). The authors were especially interested in plausibility and usefulness of the three levels.

In our research we use OHSUMED, a corpus subset of the MEDLINE database. This collection has been also used by many researchers to perform classification using MeSH keywords as class labels. However, very few have used the entire set of MeSH categories. In [6], authors limited the category pool based on the number of occurrences of these categories in the OHSUMED collection. They set this limit to 75 occurrences and used instance-based learning along with retrieval feedback to assign documents to MeSH categories. Most researchers, like in [7], [8], [9], or [10], have reduced the dataset to the documents assigned to a particular branch in the MeSH tree, namely *Heart Diseases*.

The proposed system performs automated assignment of MeSH keywords for a given medical article reference based on associative classification, which is a data mining method derived from association rule mining [11]. Since associative classification was introduced in [12], several other techniques have been presented such as classification based on multiple association rules (CMAR) [13], classification based on predictive association rules (CPAR) [14], association rule-based classification with all categories (ARC-AC), and association rule-based classification by category (ARC-BC) [15]. The latter technique describes a method of mining rules for each class separately. This approach has been used to build ACRI [16], the tool modified and applied in the proposed system. In [17] the authors employed recursive learning into associative classification and created a classifier capable of assigning a ranked list of classes to each instance. As opposed to this solution where the final set of classes is not explicitly specified, our system is capable of selecting a certain number of classes equal or close to a real number of classes that should be assigned to a given document.

Understanding the Data

An example of an article reference from the MEDLINE database is shown in Figure 1. Each article (in our work here often referred as document) consists of several descriptive attributes such as a unique identifier, author, title, journal information, and abstract. A set of keywords (marked as *M* in the figure) is used to describe the document. These keywords are manually selected among the headings from the MeSH database.

MeSH is a controlled vocabulary thesaurus of medical terms consisting of over 22,000 descriptors arranged in an 11-level hierarchical structure. The structure constitutes a tree with 15 root concepts, such as *Anatomy*, *Organisms*, or *Diseases*, that successively branch into more specific concepts. Although the concepts (other than those at the top of the hierarchy) can occur more than once in the tree, each of them has its own unique identifier.

The experimental evaluation of the proposed system is performed using a standard subset of MEDLINE, called OHSUMED [1], which consists of articles limited to 5 years (1987 to 1991). This collection contains a total of 348,543 articles.

Most researchers who used the OHSUMED have been further limiting this collection to the documents assigned to the total of 119 keywords of the *Heart Diseases* MeSH branch [8]–[10]. In this article we consider the whole spectrum of MeSH categories generalized to the second level of the tree. Generalization in this case means replacing an original MeSH keyword with the keyword located at least one level higher in the tree hierarchy. Although the second-level generalization results in the number of categories which is comparable to the *Heart Diseases* number of categories, it modifies the problem as follows:

1) As opposed to the *Heart Diseases* branch, where the majority of documents are assigned to only one category, the second-level generalization yields on average around ten categories assigned to a single document. Document-category distribution for these two cases is shown in Figure 2.

.I54711 .U 88000001 .S Alcohol Alcohol 8801; 22(2): 103-12	
.M Acetaldehyde/*ME; Buffers; Catalysis; HEPES/PD; Nuclear Magnetic Resonance; Phosphates/*PD; Protein Binding; Ribonuclease, Pancreatic/Al/*ME; Support, U.S. Gov't, Non-P.H.S.; Support, U.S. Gov't, P.H.S.	①
.T The binding of acetaldehyde to the active site of ribonuclease: alteration in catalytic activity and effects of phosphate	②
.P JOURNAL ARTICLE	
.W Ribonuclease A was reacted with [1-13C,1,2-14C]acetaldehyde and sodium cyanoborohydride in the presence or absence of 0.2 M phosphate. After several hours of incubation at 4 degrees C (pH 7.4) stable acetaldehyde-RNase adducts were formed, and the extent of their formation was similar regardless of the presence of phosphate, [...]	②
.A Mauch TJ; Tuma DJ; Sorrell MF.	

Fig. 1. Example of a MEDLINE article from the OHSUMED corpus.

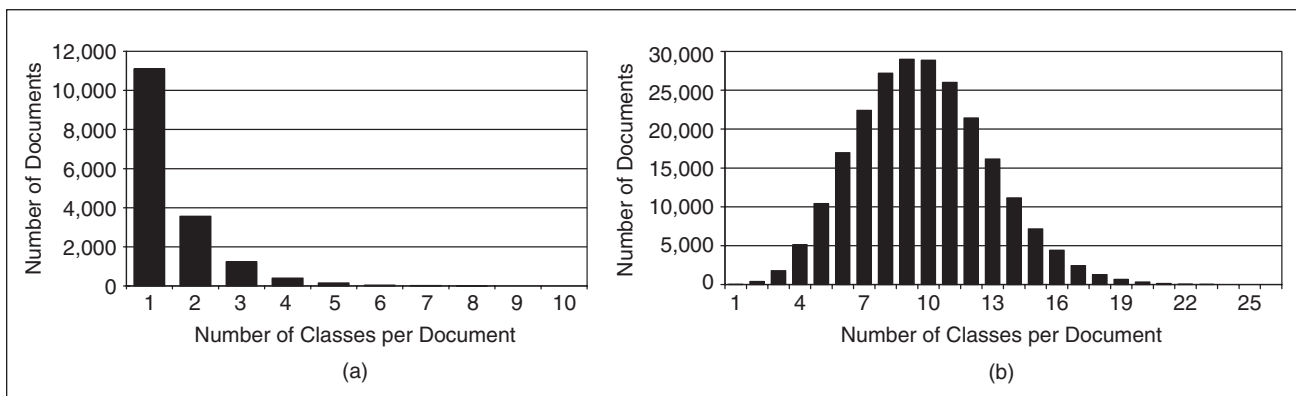


Fig. 2. Document distribution over classes for (a) the *Heart Diseases* branch and (b) a second-level generalization of OHSUMED.

2) Narrowing the MeSH tree to one of its branches limits the number of documents to those fitting this branch. Yet generalization limits only the number of categories, and the total number of documents remains unchanged. As a result, there are over 233,000 documents used in experiments instead of the 16,000 records the *Heart Diseases* branch consists of.

Data Preparation

From Article to Transaction

Associative classification, described in more detail below, requires a specific form of documents, called a transaction, for both learning and classification. Each article is seen by associative classification as a set of words drawn from the article's title, abstract, and MeSH keywords. These fields are marked by circled numbers "1" and "2" in Figure 1 (the same numbers correspond to the numbers in Figure 3 and Figure 4). The title and abstract are merged together and are separated from MeSH keywords. A transaction, which is an associative classification form of a document, consists of two parts: a set of keywords and a set of words extracted from a title and an abstract. The process of transforming a document into a transaction is shown in Figure 3. Figure 4 shows the transformation of the document given in Figure 1 by a sequence of processing blocks shown in Figure 3.

After the three fields from an original document are selected, the title and abstract are processed separately from the MeSH keywords. The keywords are normalized using the MeSH tree identifiers as shown in Box 3 of Figure 4. Each identifier consists of a series of numbers (separated by a dot) being the consecutive identifiers of branches of the MeSH tree from the root node to the appropriate node indicated by an input keyword. The only exception is the very first part of the identifier, e.g., D02, which indicates two levels at once: one denoted by a letter and the other denoted by a remaining number. Therefore, to obtain generalization to the second level the identifiers need to be contracted to the first part only, the result of which is shown in Box 4 of Figure 4.

The title and abstract of the considered document are also normalized. This normalization consists of stop word pruning and word stemming. Stop words are words that appear frequently but are irrelevant with respect to classification (e.g., a, the, of, at, in, etc.) and therefore are not included in the transaction. Word stemming aims to unify the morphological variants of the word, which in practice boils down to removal

of word endings. We employed the widely used Porter's algorithm [18] to perform this operation. The result of the normalization process is shown in Box 5 of Figure 4.

The last step is to combine both keyword identifiers generalized to the second level and the normalized title and abstract into a single transaction as shown in Box 6 of Figure 4. The second-level generalization resulted in 114 categories (classes) distributed as shown in Figure 2(b).

Dataset Split

Due to lack of an abstract for some of the documents from the OHSUMED collection, we reduced this collection to documents that have both a title and an abstract, which resulted in a total of 233,445 documents. The collection was divided into two subsets: 1) 183,229 documents dated 1987 through 1990, which were used as a training set, and 2) 50,216 documents dated 1991, which were used as a testing set. This split conforms to the work of other researchers [8]–[10]. However, unlike the others who used the testing set to tune the

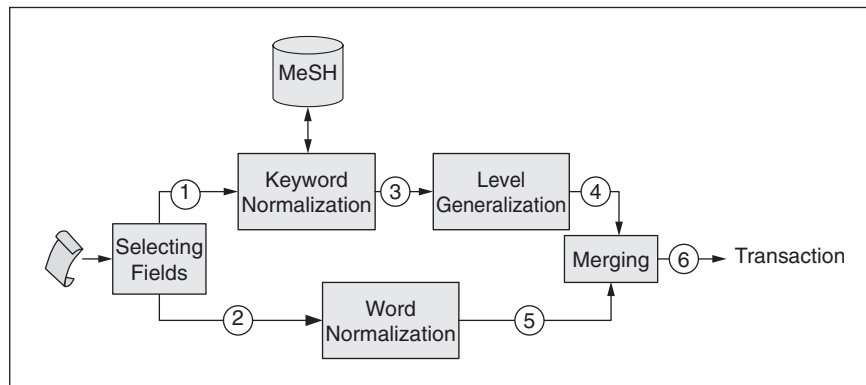


Fig. 3. Document preprocessing.

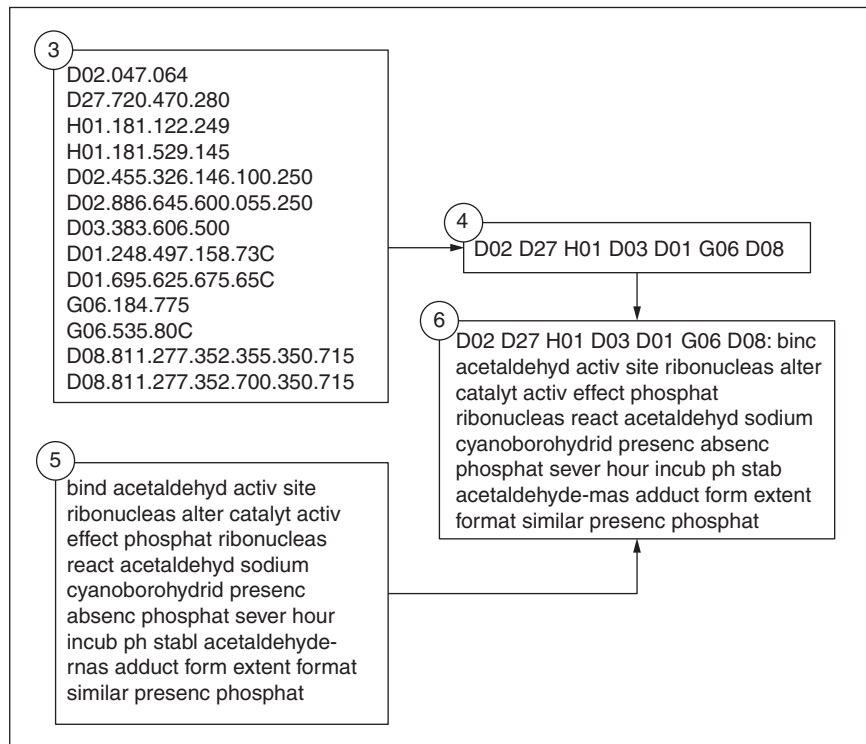


Fig. 4. Document transformation.

**MeSH is a controlled vocabulary
thesaurus of medical terms consisting of over
22,000 descriptors arranged in an 11-level
hierarchical structure.**

parameters of a classification process (for more details, see the System Overview section), we performed ten-fold cross validation on the training set (the ten-fold cross validation technique requires the training set to be further divided into training and validation sets) and used the obtained parameters to validate the classification system with the testing set. Although this tuning possibly results in a lower performance compared with performance of methods tuned on the testing set, it describes performance of classification models that are not overfitted to the testing set. A similar approach has been used in [6] but limited to two training subsets only.

Data Mining

From Research Goals to Data Mining Goals

The original goals that aim at development of an automated system for assigning MeSH keywords to MEDLINE articles are translated into the corresponding data mining goals:

- 1) As opposed to multiclass classification, our objective is to build a multilabel classifier. Therefore, the proposed system has to be capable of dealing not only with multiple classes but also with assigning a various number of classes to a single object.
- 2) We extend original associative classification to a classification that considers reoccurrence of words in a single transaction. This method is further compared to the approach that does not consider reoccurrence of words.

Associative Classification

Associative classification is a process of assigning labels to objects based on rules obtained during the association rule mining process. Association rules were originally introduced

in [11] and further extended to class association rules (CARs) [12] that are directly used in associative classification. The main idea of associative classification is to extend the original structure of transactions known from association rules mining (i.e., a set of items) by adding a class label to each transaction. Items are uniquely identified features describing each object and may have different representation depending on an application. In our case, items represent words in documents.

The association rule mining process results in a set of rules in the form of $condition \Rightarrow class$, where $condition$ is a set of items. Rules are generated based on user-defined minimum support and minimum confidence, which are the conventional parameters used in association rule mining [11]. Generated rules are used by a classification system to predict the class of new objects.

To perform experiments we employed the associative classifier with reoccurring items (ACRI), a tool for associative classification that considers reoccurrence of items in a single transaction during both generation of association rules and classification [16]. Recurrent items were originally described in [19]. Taking recurrent items into consideration during association rule mining results in an altered form of the condition part of a rule that is now enriched by the number of the occurrences of each word.

Although ACRI is developed to be used with recurrent items, it is possible to use it also as a simple nonrecurrent-item-based classifier. The differences in performance between those two types of classifiers are compared in this article.

Single-Versus Multilabel Classification

In a single-label classification problem, each document is labeled with one class only. When two or more rules match a document (i.e., words in a document match words in a rule) usually the best rule (e.g., the one with the highest confidence) is selected and used to classify the document while the remaining rules are discarded. However, multilabel classification assigns more than one class to each document. Assuming that rules are in the form of $condition \Rightarrow class$, more than one rule needs to be used to assign possibly more than one label per document as the result of classification. This requires development of rule ranking and rule elimination techniques, which are described in the next section.

System Overview

The associative classification for MEDLINE documents is shown in Figure 5.

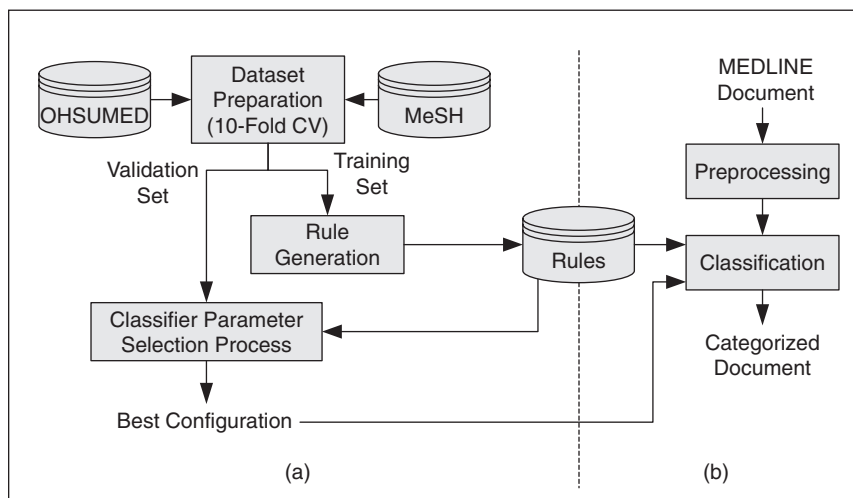


Fig. 5. Process of (a) building a classifier and (b) classification.

The learning process [i.e., the process of building the classifier shown in Figure 5(a)] is performed to, first, generate a set of association rules and, second, to tune a set of the algorithm's parameters to classify new (unseen) data with the highest accuracy. Generated rules together with the best parameter configuration are used to classify new MEDLINE articles, as shown in Figure 5(b).

The rule-generation step generates frequent item sets between the article's words and the corresponding MeSH keywords, resulting in a set of rules.

Although original association rule mining takes two parameters, namely, support and confidence thresholds, preliminary experiments showed that only minimum confidence has a significant effect on the quality of classification. At the same time, the number of generated rules is controlled using the support threshold. Therefore, minimum support is used during the rule-generation step to initially reduce the number of rules, which is further decreased during classification by the confidence threshold.

During the tuning step a number of algorithm's parameter values are tested to discover an optimal set of values used further in classification. This set of parameters is responsible for pruning, ranking, and selection of rules. By pruning we mean the process of reducing the number of rules needed for the classification. Rule ranking is needed to choose more than one rule to classify a document. Uneven distribution of classes over documents in the dataset [see Figure 2(b)] requires also a technique to select the proper number of previously ranked rules.

The pruning process is based on one parameter (minimum confidence), whereas ranking and selection can be performed in many ways. We consider three possibilities: 1) using confidence to prune rules, 2) using confidence to both prune and rank rules, and 3) using the cosine of an angle between rules and documents to prune and rank rules. This leads to the following five configurations:

- **Simple (R_{sim}).** Rules are only pruned (based on minimum confidence). Ranking and selection are not performed.
- **Confidence factor (R_{con}).** Rule ranking is based on the rules' confidence. Selection is parameterized by selection factor, a value denoting the percentage of rules that should remain after this operation.
- **Cosine factor (R_{cos}).** Rule ranking is based on the cosine measure between a rule and a document (represented by words). Cosine measure is a value equal to the angle between two vectors. The position of vectors in n -dimensional space is indicated by the number of reoccurrences of corresponding words. Selection is performed using selection factor exactly as in the R_{con} configuration.
- **Simple, nonrecurrent (S_{sim}).** This is similar to the simple configuration but rules are generated without considering the frequency of words in a document.
- **Confidence factor, nonrecurrent (S_{con}).** This is the same as the confidence factor configuration but using nonrecurrent-item-based rules.

Since nonrecurrent-item based classification does not carry information about the occurrences of words in a document, we do not consider cosine factor configuration in this case. R -configurations and S -configurations are used to denote a set of configurations considering reoccurrence of items and a set of configurations that neglects this information, respectively.

In the final step of classification, a set of classes is predicted by combining the classes that appear as a consequence in the set of selected rules. An example of classification using the five configurations is shown in Figure 6. For example, the configuration R_{sim} with minimum confidence of 50% selects three rules ($R1$, $R2$, and $R3$) out of the four matching the input documents and assigns two classes: $C1$ (both $R1$ and $R2$ imply the same class $C1$) and $C2$ (implied by $R3$) to the document.

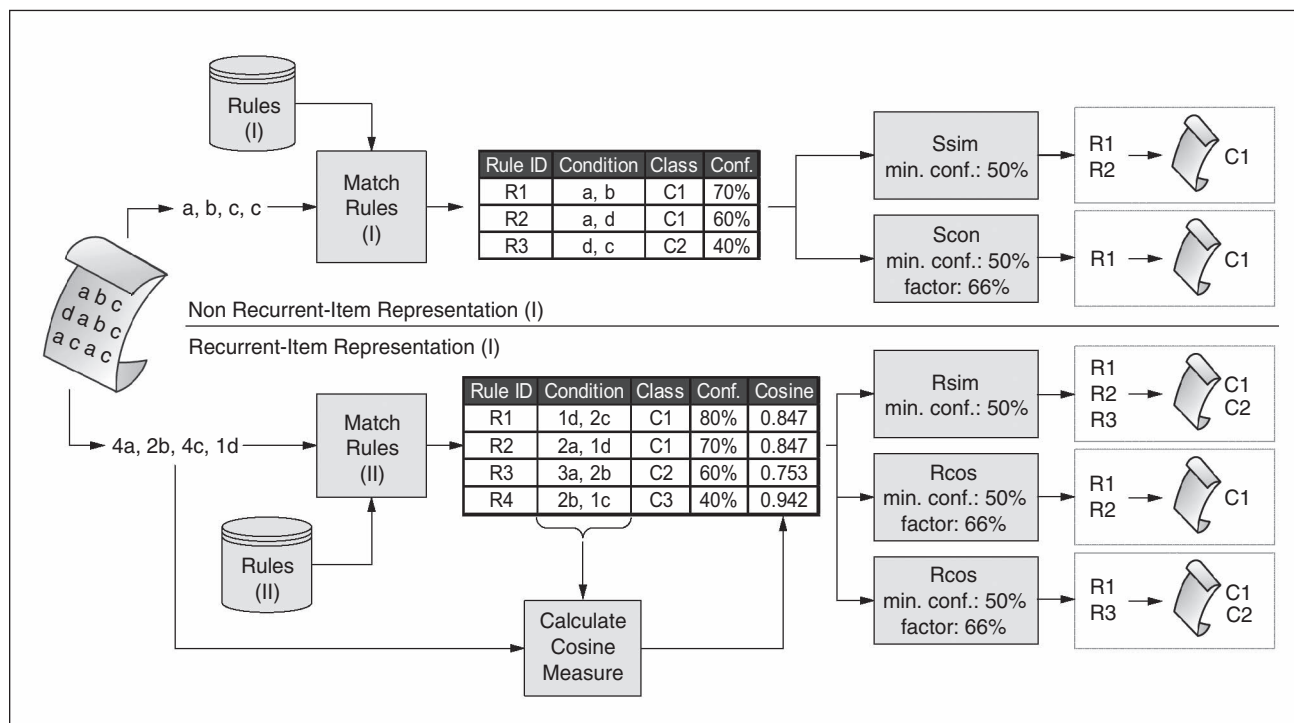


Fig. 6. Example of classification.

Each article is seen by associative classification
as a set of words drawn from the article's title,
abstract, and MeSH keywords.

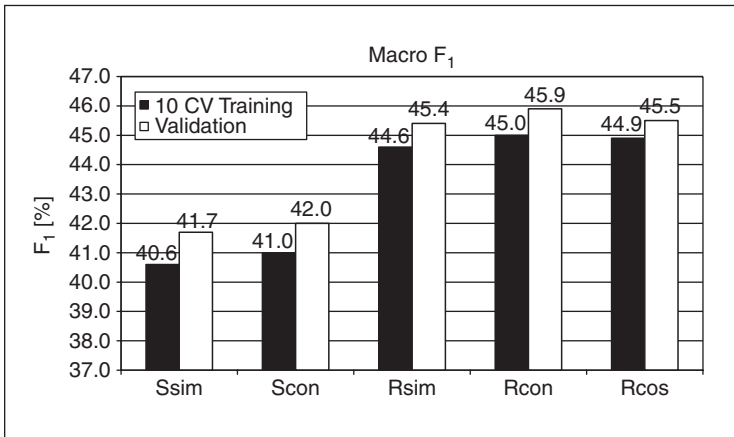


Fig. 7. Best macro F_1 .

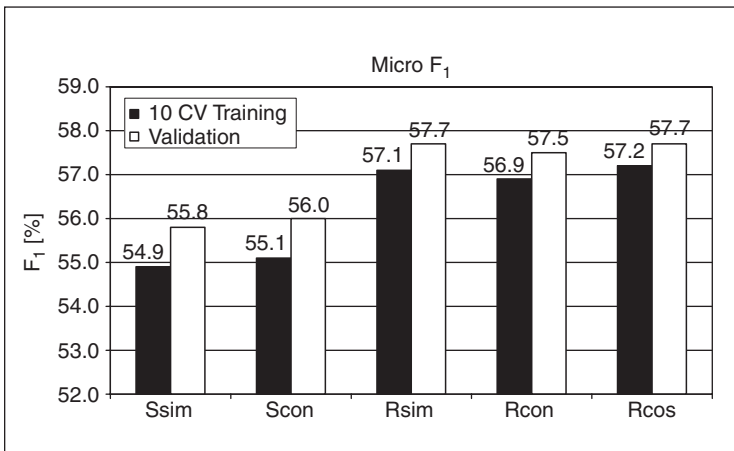


Fig. 8. Best micro F_1 .

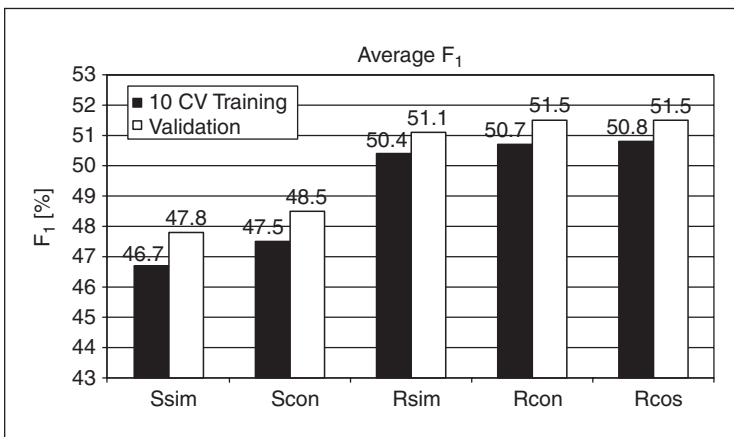


Fig. 9. Best average F_1 .

Evaluation of Discovered Knowledge

Evaluation Criteria

Evaluation of classification quality is based on commonly used measures, such as precision (p), recall (r), and the combination of the two (F_1) [20].

Unlike single-class classification, multiclass classification requires averaging individual results from contingency matrices built for each class. We report two types of averaging: macro-averaging and micro-averaging [21]. Macro-averaging emphasizes the ability of a classification system to behave well for all classes, even those with a low number of examples (documents), whereas micro-averaging reflects better a classification for larger classes with the expense of poorer results for small classes. In this article most consideration is given to F_1 , which is commonly used in text categorization, but we also report on precision and recall to give further insights.

Results

Searching through the space of classification parameters for all five configurations took approximately 170 ten-fold cross-validation experiments. Parameters used in the learning process were tuned to maximize the value of macro F_1 , micro F_1 , and the average of those two. Thus, the experiments performed on the training set resulted in selection of three sets of parameters for each configuration. Tuning the algorithm for the three values of F_1 allows for comparing differences between micro- and macro-averaging. Results for micro, macro, and average F_1 for both training and testing are presented in Figure 7–Figure 9. Due to space limitations we show only F_1 as the most representative and most often used measure of classification quality in text categorization. The test performed with the same 15 sets of parameters for the testing set showed that the order of the configurations with respect to F_1 for each considered averaging method is consistent with the results obtained with the training dataset, which validates the correctness of the chosen parameters.

For macro-averaging, the best results are obtained for R_{con} , whereas R_{cos} works better with micro-averaging and in the case of optimizing the average between macro- and micro-average. This is consistent for both cross validation performed on training data and tests run on an independent test set. The parameters resulting from optimization of the algorithm for each of the considered measures are shown in Table 1. Parameters were tuned with a 10% step.

Table 1. Best configuration parameters with respect to optimized measure.

Configuration	Macro F_1		Micro F_1		Average F_1	
	Min. conf. (%)	Factor (%)	Min. conf. (%)	Factor (%)	Min. conf. (%)	Factor (%)
R_{sim}	50		60		60	
R_{con}	30	40	50	60	40	50
R_{cos}	40	70	60	90	50	80
S_{sim}	40		50		40	
S_{con}	30	70	40	60	30	50

Except for one set of parameters (micro F_1 and average of micro and micro F_1 for R_{sim}), no similarity is found between measures for a particular configuration. This implies that choosing appropriate configuration and parameters is dependent on the type of a considered measure.

Although differences in performance among R -configurations and S -configurations are rather small, the results indisputably show that configurations utilizing the knowledge of occurrences of words in a document are better than those that neglect this information. This difference is especially visible in macro F_1 scores, which reaches around 4%. The best results are 46% and 42% for a configuration considering reoccurrence of items and configuration that do not take this information into account, respectively. A similar difference is also observed for both micro F_1 and average F_1 .

The number of rules after pruning used in classification performed on the ten-fold cross-validation training set for each configuration and type of averaging is shown in Figure 10. Although R -configurations perform better than S -configurations they require more rules. Making further comparison, performance with respect to F_1 follows the trend of the number of rules needed for classifiers optimized based on macro F_1 . However, for micro and average F_1 this is not the case. Although R_{con} classifies worse than both R_{cos} and R_{sim} , it requires more rules than the latter two when considering micro F_1 and average F_1 .

The average time for classifying the training set for each configuration, shown in Figure 11, follows, with one exception, the number of rules needed to perform this operation. Analysis of Figures 10 and 11 leads to the conclusion that when optimizing the classifier for best macro F_1 it may be worth considering choosing R_{cos} instead of R_{con} , which gives slightly better accuracy with the expense of the significantly longer training time.

The distribution of documents from the testing set with respect to F_1 , precision, and recall for the three best configurations that correspond to micro, macro, and average F_1 is shown in Figure 12. The distribution over the F_1 score for the best micro-average configuration is slightly flatter than the configuration for the best macro-average. An even bigger difference is observed for the distribution over precision and recall. Optimizing macro F_1 results in a greater number of documents classified with high precision but, at the same time, a low number of documents is classified with high recall. An exactly opposite situation is observed when optimizing micro F_1 .

Precision is inversely proportional to FP examples and thus its low values show a tendency to “overclassify”; i.e., to assign a single document to additional incorrect categories. In other words, the classification model is too general. On the other hand, recall is inversely proportional to FN examples, reflecting an inability of assigning classes to a document. This means that the model is too specific. Observations based on Figure 12 are consistent with the definition of macro- and micro-averaging. Micro-average is a weighted average and thus is more suitable when a user is interested in maximizing performance for categories with a large number of examples, neglecting, to a certain extent, the classification accuracy of categories with a relatively low number of examples.

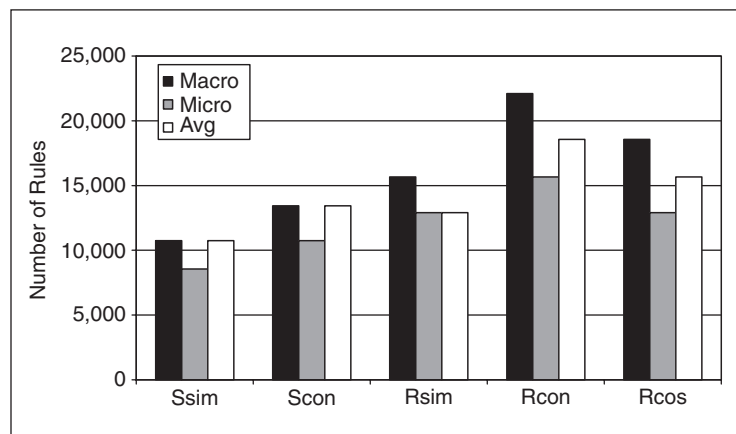


Fig. 10. Number of rules after pruning used with a ten-fold cross-validation training set for each configuration.

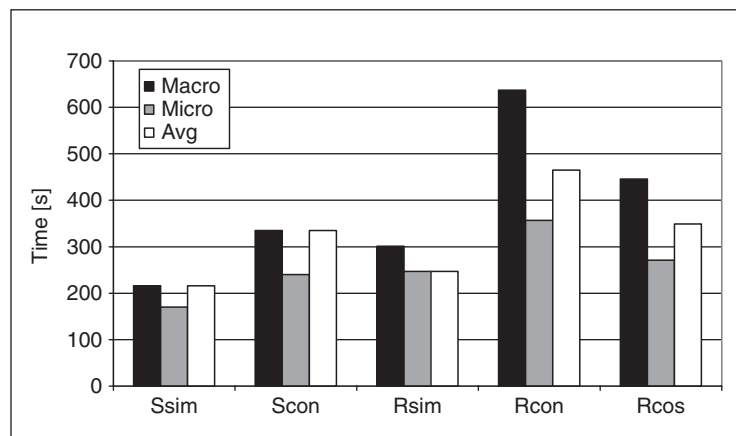


Fig. 11. Average time needed to accomplish classification on a ten-fold cross-validation training set.

A relatively small amount of research has been dedicated to investigation of multilabel classification, especially using associative classification.

To the best of our knowledge no one has tried to generalize the MeSH tree in a manner described in this article. Only a few researchers have tested their text categorization systems using the entire set of MeSH keywords and all instances from the OHSUMED dataset. Among them, the closest are results presented in [6]. The authors obtained a macro F_1 score of 44% using MeSH categories with at least 75 examples (documents) in the training set.

Using the Discovered Knowledge

The experimental results are used to put together three scenarios a user may follow when applying the proposed system.

- 1) When parameters corresponding to optimization of micro F_1 are used, the model will perform with higher precision and lower recall. This means that although a document will be classified to correct categories, some of the categories may be omitted.
- 2) When parameters corresponding to optimization of macro F_1 are used, higher recall and lower precision will be achieved. Although in this case fewer categories will be omitted, a larger number of incorrect predictions will be made.
- 3) Using the average of macro and micro F_1 results in the tradeoff between the above two situations.

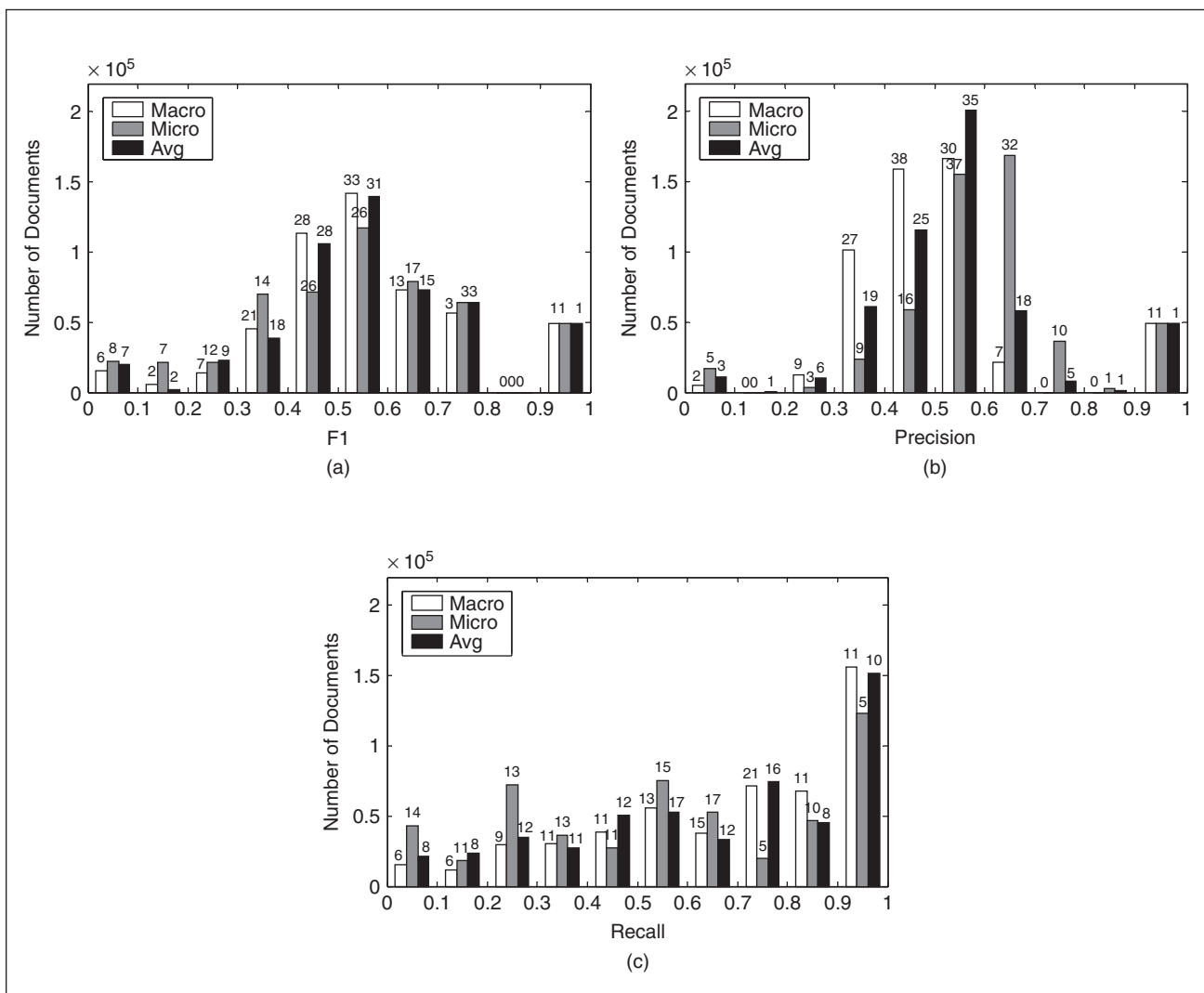


Fig. 12. Document distribution with respect to F_1 , precision, and recall. The numbers above the bars indicate the number of classes.

Summary and Conclusions

The specific characteristic of classification of medical documents from the MEDLINE database is that each document is assigned to more than one category, which requires a system for multilabel classification. Another major challenge was to develop a scalable method capable of dealing with hundreds of thousand of documents. We proposed a novel system for automated classification of MEDLINE documents to MeSH keywords based on the recently developed data mining algorithm called ACRI, which was modified to accommodate multilabel classification. Five different classification configurations in conjunction with different methods of measuring classification quality were proposed and tested. The extensive experimental comparison showed superiority of methods based on reoccurrence of words in an article over nonrecurrent-based associative classification. The achieved relatively high value of macro F_1 (46%) demonstrates the high quality of the proposed system for this challenging dataset. Accuracy of the proposed classifier, defined as the ratio of the sum of TP and TN examples to the total number of examples, reached 90%.

Three scenarios were proposed based on the performed tests and different possible objectives. If a goal is to classify the largest number of documents, a configuration that maximizes micro F_1 should be chosen. On the other hand, if a system is to work well for categories with a small number of documents, a configuration that maximizes macro F_1 is more suitable. A tradeoff can be obtained by using a configuration that optimizes the average between macro and micro F_1 .



Rafal Rak received his M.Sc. with honors in automation and robotics at the AGH University of Science and Technology in Krakow, Poland, in 2003. He was involved in research projects related to software engineering with special emphasis on software project management in a distributed environment. He has three years of professional

experience in industry. Since September 2004, he has been a Ph.D. student in the Department of Electrical and Computer Engineering at the University of Alberta, in Edmonton, Alberta, Canada. His work on associative classification resulted in several publications. His research interests include machine learning, knowledge discovery, and semantic data analysis.



Lukasz A. Kurgan received his M.Sc. with honors (and was recognized with an Outstanding Student Award) in automation and robotics from the AGH University of Science and Technology, Krakow, in 1999, and his Ph.D. in computer science from the University of Colorado at Boulder, in 2003. He is currently an assistant professor in the

Department of Electrical and Computer Engineering at the University of Alberta in Edmonton, Alberta, Canada. His research interests include data mining and knowledge discovery, machine learning, computational biology, and bioinformatics. He currently serves as an associate editor of *Neurocomputing*. Dr. Kurgan is a member of the IEEE, ACM, and ISCB.

Marek Reformat received his M.Sc. (with honors) from Technical University of Poznan, Poland, and his Ph.D. from University of Manitoba, Winnipeg, Manitoba, Canada.



Currently, he is with the Department of Electrical and Computer Engineering at the University of Alberta, Edmonton, Alberta, Canada. His research interests are in the areas of application of computational intelligence techniques, such as neurofuzzy systems and evolutionary computing, as well as probabilistic and evidence theories to intelli-

gent data analysis and modeling leading to translating data into knowledge. He applies these methods to conduct research in the areas of software and knowledge engineering. Dr. Reformat has been a member of program committees of several conferences related to computational intelligence and evolutionary computing. He is a member of the IEEE and ACM.

Address for Correspondence: Rafal Rak, Department of Electrical and Computer Engineering, University of Alberta, 9107-116 Street, Edmonton, Alberta, T6G 2V4, Canada. E-mail: rrak@ece.ualberta.ca.

References

- [1] C. Hersh, C. Buckley, T.J. Leone, and D. Hickam, "OHSUMED: An interactive retrieval evaluation and new large test collection for research," in *Proc. 17th Annu. Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, 1994, pp. 192-201.
- [2] R. Rak, L. Kurgan, and M. Reformat, "Multi-label associative classification of medical documents from MEDLINE," in *Proc. 4th Int. Conf. Machine Learning and Applications*, Los Angeles, CA, 2005, pp. 177-184.
- [3] N. Uramoto, H. Matsuzawa, T. Nagano, A. Murakami, H. Takeuchi, and K. Takeda, "A text-mining system for knowledge discovery from biomedical documents," *IBM Syst. J.*, vol. 43, no. 3, pp. 516-533, 2004.
- [4] P. Kankar, S. Adak, A. Sarkar, K. Murali, and G. Sharma, "MedMeSH summarizer: Text mining for gene clusters," in *Proc. 2nd SIAM Int. Conf. Data Mining*, Arlington, VA, 2002, pp. 548-565.
- [5] C. Blake and W. Pratt, "Better rules, few features: A semantic approach to selecting features from text," in *Proc. IEEE Int. Conf. Data Mining*, San Jose, CA, 2001, pp. 59-66.
- [6] W. Lam, M. Ruiz, and P. Srinivasan, "Automatic text categorization and its application to text retrieval," *IEEE Trans. Knowledge Data Engineering*, vol. 11, no. 6, pp. 865-879, 1999.
- [7] W. Lam and C.Y. Ho, "Using a generalized instance set for automatic text categorization," in *Proc. 21st Annu. Int. Conf. Research and Development in Information Retrieval*, Melbourne, Australia, 1998, pp. 81-89.
- [8] D.D. Lewis, R.E. Schapire, J.P. Callan, and R. Papka, "Training algorithms for linear text classifiers," in *Proc. 19th Annu. Int. Conf. Research and Development in Information Retrieval*, Zurich, Switzerland, pp. 298-306, 1996.
- [9] Y. Yang, "An evaluation of statistical approaches to MEDLINE indexing," in *Proc. Conf. American Medical Informatics Association*, Washington, D.C., 1996, pp. 358-362.
- [10] M.E. Ruiz and P. Srinivasan, "Hierarchical text categorization using neural networks," *Information Retrieval*, vol. 5, no. 1, pp. 87-118, Jan. 2002.
- [11] R. Agarwal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. Int. Conf. Very Large Databases*, Santiago de Chile, Chile, 1994, pp. 487-499.
- [12] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," in *Proc. Knowledge Discovery Data Mining*, New York, NY, 1998, pp. 80-86.
- [13] W. Li, J. Han, and J. Pei, "CMAR: Accurate and efficient classification based on multiple class-association rules," in *Proc. IEEE Int. Conf. Data Mining*, San Jose, CA, 2001, pp. 369-376.
- [14] X. Yin and J. Han, "CPAR: Classification based on predictive association rules," in *Proc. 3rd SIAM Int. Conf. Data Mining (SDM'03)*, San Francisco, CA, 2003, pp. 369-376.
- [15] O.R. Zaiane and M.-L. Antonie, "Classifying text documents by associating terms with text categories," in *Proc. 13th Australasian Database Conf.*, Melbourne, Australia, 2002, pp. 215-222.
- [16] R. Rak, W. Stach, O.R. Zaiane, and M.-L. Antonie, "Considering re-occurring features in associative classifiers," *Lecture Notes Comput. Sci.*, vol. 3518, pp. 240-248, 2005.
- [17] F. Thabtah, P. Cowling, and Y. Peng, "MMAC: A new multiclass, multi-label associative classification approach," in *Proc. 4th IEEE Int. Conf. Data Mining*, Brighton, England, 2004, pp. 217-224.
- [18] M.F. Porter, "An algorithm for suffix stripping," in *Readings Information Retrieval*, San Mateo, CA: Morgan Kaufman, 1997, pp. 313-316.
- [19] O. Zaiane, J. Han, and H. Zhu, "Mining recurrent items in multimedia with progressive resolution refinement," in *Proc. 16th Int. Conf. Data Engineering*, 2000, pp. 461-470.
- [20] C.J. van Rijsbergen, *Information Retrieval*, 2nd ed. London, UK: Butterworth, 1979.
- [21] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1-47, 2002.