Multi-label Associative Classification of Medical Documents from MEDLINE

Rafal Rak
rrak@ece.ualberta.ca

Lukasz Kurgan
lkurgan@ece.ualberta.ca

Marek Reformat
reform@ece.ualberta.ca

University of Alberta, Electrical and Computer Engineering
9107 - 116 Street, Edmonton, Alberta, Canada T6G 2V4

Abstract

Ability to provide convenient access to scientific documents becomes a difficult problem due to large and constantly increasing number of incoming documents and extensive manual work associated with their storage, description and classification. This requires intelligent search and classification capabilities for users to find required information. It is especially true for repositories of scientific medical articles due to their extensive use, large size and number of new documents, and well maintained structure. This research aims to provide an automated method for classification of articles into the structure of medical document repositories, which would support currently performed extensive manual work. The proposed method classifies articles from the largest medical repository, MEDLINE, using state of the art data mining technology. The method is based on a novel associative classification technique which considers recurrent items and most importantly multi-label characteristic of the MEDLINE data. Based on large scale experiments that utilize 350,000 documents several different classification algorithms have been compared including both recurrent and non-recurrent associative classification. The algorithms are capable of assigning each medical document to several classes (multi-label classification) and are characterized by relatively high accuracy. We also investigate different measures of classification quality and point out pros and cons of each. Based on experimental result we show that recurrent item based associative classification demonstrates superior performance and propose three alternative setups that allow the user to obtain different desired classification qualities.

1. Introduction

Convenient and timely access to scientific documents becomes a significant problem due to constantly increasing number of new documents and necessity to provide intelligent search and categorization capabilities. In this paper, we focus on categorization of medical journal documents from the MEDLINE database. MEDLINE is the National Library of Medicine's (NLM) database consisting of approximately 13 million article references to biomedical journal articles dated back to 1966. The database is rapidly expanding, with over 500,000 new article references added every year, which translates to about 1500-3500 references per day. The NLM employees manually assign Medical Subject Headings (MeSH), the NLM's controlled vocabulary thesaurus consisting of medical terms at various levels of specificity [11], to each incoming article reference. The exponentially growing number of incoming articles combined with the error prone nature of human work make this task very difficult. This paper describes development of a system for automated classification of MEDLINE article references. We employ and redesign a recently developed data mining based classification tool to assign MeSH keywords to article references. The method is designed and tested on the OHSUMED corpus [4], a comprehensive set of almost 350,000 article references, performs challenging multi-class multi-label classification procedure, and applies state of the art associative classification. The developed method is extensively tested employing five different approaches to multi-label associative classification problem in order to choose optimal configuration. We also investigate several different measures of classification quality that result in alternative setups and different performance characteristics.

2. Background

2.1. Related work

Significant amount of attention was already devoted to research and utilization of NLM's tools and databases. Uramoto et al. developed MedTAKMI (Text Analysis and Knowledge MIning for Biomedical Documents), an application to facilitate knowledge discovery from very large text databases, such as MEDLINE [17]. The application dynamically mines documents to obtain their

Proceedings of the Fourth International Conference on Machine Learning and Applications (ICMLA'05)
0-7695-2495-8/05 $20.00 © 2005 IEEE

characteristic features. It uses categories such as MeSH keywords for term extraction and interactive series of drill-down queries. The 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 the gene in a user-defined query [5]. Blake and Pratt explored relationship between features used to represent text and the quality of the final model for medical texts with application to MEDLINE [3]. They used association rules to compare three different semantic levels: words, MeSH keywords, and automatically selected concepts coming from NLM's Unified Medical Language System (UMLS). They were especially interested in plausibility and usefulness of the three levels.

The OHSUMED corpus as a subset of the MEDLINE database have been used by many researchers to perform classification using MeSH keywords as class labels. Most of them has reduced the dataset to documents assigned to a particular subtree, *Heart Diseases*, in the MeSH tree (e.g. [6], [8], [19], [14]). Very few however, used the entire set of MeSH categories. Lam et al. [7] limited the category pool to those that occur more than 75 times in the OHSUMED dataset. They used *instance-based learning* and *retrieval feedback* to assign documents to MeSH categories.

The proposed system uses associative classification, method based on association rules [1]. Its main advantage is an ability to work with large datasets, which stems from scalability of the association rule generation techniques. Several associative classification systems have been introduced so far, such as CBA [10], CMAR [9], CPAR [20], ARC-AC, and ARC-BC [22]. The latter two were used to classify medical images [2], while ARC-BC approach was used to build ACRI [13], the tool modified and applied in the proposed system.

The considered problem of classification of medical documents is characterized by multi-labeling. Each document is assigned to several classes, and thus classification method capable of dealing with this situation is required. Thabtah et al. introduced several different classification quality measures for multi-label classification problem [16]. They employed *recursive learning* into associative classification and created a classifier capable of assigning a ranked list of classes (i.e. the final set of classes is not explicitly specified) to each instance. In contrast, 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.

2.2. MEDLINE and MeSH

Each article reference in MEDLINE database includes information such as a unique identifier, author, title, journal information, abstract, and 10 to 15 manually assigned MeSH keywords. Among this information, a title and abstract have been chosen as an input to our system. We refer to them simply as document. MeSH is an annually updated controlled vocabulary thesaurus of medical terms [11]. Currently there are over 22,000 *descriptors* arranged in an 11-level hierarchical structure. At the top of the tree structure there are 15 general concepts (keywords) such as *Anatomy*, *Organisms*, or *Diseases*. Any other lower-level keyword can occur more than once in a tree. A fragment of the MeSH tree is shown in Figure 1.

2.3. Associative classification

The associative classification, i.e. classification that uses rules obtained from an association rule mining process to classify objects, was originally introduced as Class Association Rules (CAR) [10]. The main idea of CAR was to extend the structure of transactions, known from association rules mining [1], by adding a class label to each transaction. A generated set of rules in form of $condset \rightarrow c$, where $condset$ is a set of items and c is a class label, is used as a classification system to predict the class of new objects. We employed ACRI (*Associative Classifier with Reoccurring Items*) tool for associative classification [13] that combines the associative classification with the recurrent items theory, originally described by Zaïane et al. [21]. Recurrent items theory modifies the original approach using transactions of the form $< i_1, i_2, ..., i_n, c >$, where i_i is an item in a transaction and c is a class label, to the form of $< o_1 i_1, o_2 i_2, ..., o_n i_n, c >$, where o_i is the number of occurrences of the item i_i in the transaction.

For the purpose of this paper the terms *document* and *word* are used as the equivalents for *transaction* and *item*, respectively.

Although ACRI is developed to use the information about occurrences of words in a document, it is possible to use it as a simple non-recurrent items based classifier. The differences in performance between those two types of classifiers are also compared in this paper.

2.4. Single- vs. multi-label classification

In single-label classification problem a document is assigned to one class only. When two or more rules match a document, i.e. words in a document comprise words in a rule, the best one (e.g. with the highest confidence) is kept while the rest is eliminated. In multi-label classification however, a document can be assigned to one or more

classes. In this case, if a rule is in the form of $condset \rightarrow c$, more than one rule need to be selected as this is the only way to eventually obtain more than one class as the output of classification.
In both cases ACRI comes with a variety of rule ranking methods some of which are utilized in this research and described in the next section.

3. Proposed approach

3.1. Motivation

Most of research based on the OHSUMED corpus were carried out using a narrowed set limited to documents assigned to the total of 119 categories of a *Heart Diseases* MeSH subtree [8] [19] [14] (see Figure 1). Our research, however, considers 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. It gives a comparable to Heart Diseases subtree number of categories; however, modifies the problem in two ways:

1. Although the *Heart Diseases* subtree is still multi-label classification problem, the majority of documents are assigned to only one category, whereas the generalization to the second level yields on average around 10 categories assigned to a single document (see Figures 2(a) and 2(b)).

2. Instead of selecting only documents that fits the subtree, the entire OHSUMED corpus is used in experiments which extends the dataset from around 16,000 records to over 233,000.

3.2. Goals

This research addresses the following goals.

1. Build a multi-class multi-label classification system based on associative classification. Although a multi-class classification problem has been widely studied, relatively small amount of research was dedicated to investigation of multi-label classification, especially using associative classification.

2. Compare several different classification approaches which are based on associative classification.

3. Employ the classification system to MEDLINE in order to support manual categorization of medical documents to the MeSH structure.

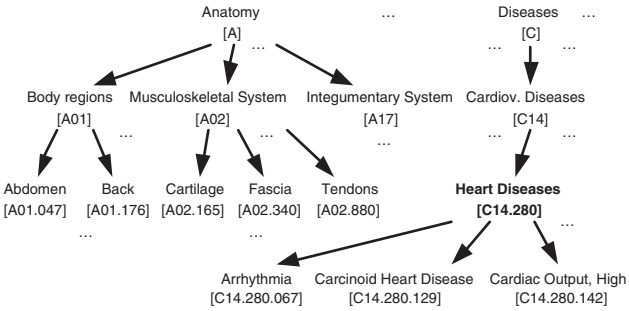


Figure 1. Fragment of MeSH tree

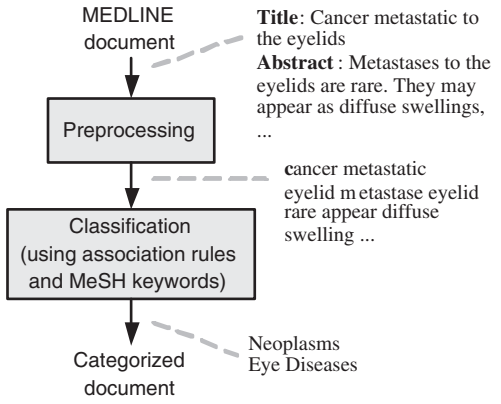


Figure 3. Classification of medical documents with the proposed system

In order to perform classification we used and modified ACRI [13], the data mining associative classification tool. The remaining of this paper is organized as follows. Section 3.3 describes the application of ACRI to build a classification system. Experimental details and results are shown in section 4. Section 5 contains discussion and summarizes the paper.

3.3. System overview

Assigning MeSH keywords to documents is depicted in Figure 3. A raw document is first preprocessed to prune unnecessary words and normalize remained words. Then associative classification is employed to assign classes to the document.
The following paragraphs briefly describe *associative classification* as well as the process of finding parameters necessary to build the classification system.

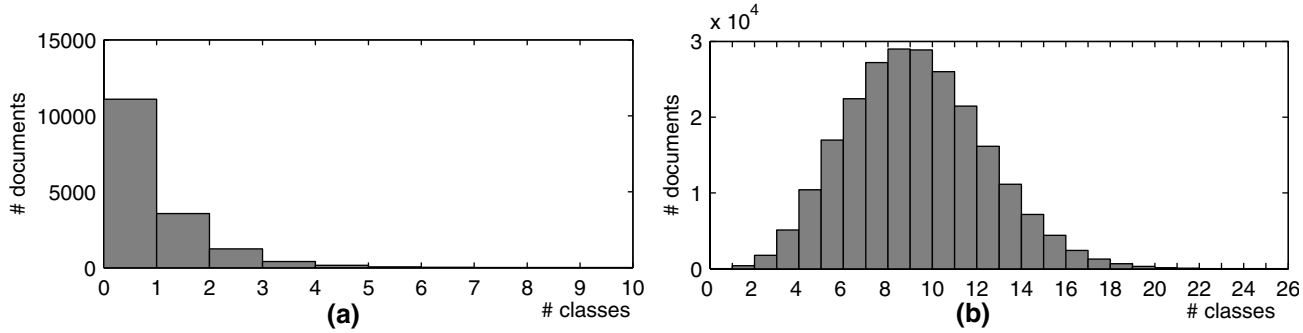


Figure 2. Document distribution over classes for (a) *Heart Diseases* **subtree and (b) the 2nd level generalization of OHSUMED**

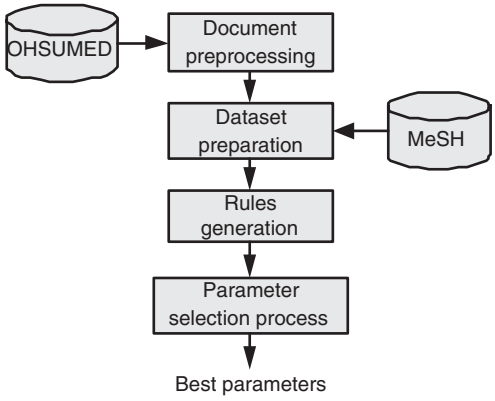


Figure 4. System design

3.4. System design

Before classification can be performed, a learning process needs to be carried out in order to find optimal values of certain parameters which are described later in this section. The learning process, in case of associative classification, is performed to (1) generate a set of association rules and (2) find the best set of parameter values that are used further in classification. The complete process diagram is shown in Figure 4.

In the *document preprocessing* stage Porter's algorithm is used to perform stemming on words as a part of the normalization of documents [12] and *stopwords* are pruned, i.e. the words that appear frequently but are irrelevant with respect to classification.

The next stage, *dataset preparation*, assigns MeSH keywords, i.e. class labels, to the documents. As it is impossible to perform classification using the entire spectrum of MeSH keywords due to their huge number (over 22,000), the keywords are generalized to the second level of the tree structured of the MeSH database as explained in Section 3.1.

Rules generation finds rules from a set of given documents of the form $< \{o_1 i_1, o_2 i_2, ..., o_n i_n\}, c >$, where o_n is the number of occurrences of word i_n and c is a class label. Rules are generated based on user-defined *minimum support* and *minimum confidence*, the conventional parameters being used in association rule mining [1]. Preliminary experiments and result analysis show that only minimum confidence has a real effect on the quality of classification, whereas minimum support is used only for the initial reduction of rules (pruned later on by confidence threshold). Rules are generated based on minimum support only, and minimum confidence is an adjustable parameter extensively exploited in the next stage.

The final stage of learning process examines a number of ACRI's parameters to discover an optimal set that can be used during classification. These parameters are responsible for *pruning*, *ranking*, and *selecting* rules which are used to classify documents. Pruning is the process of reducing a number of rules needed for further classification. Rule ranking is needed if we want to choose more than one rule to be used to classify a certain document based on the certain measure of quality. Selecting a proper number of previously ranked rules is another task in multi-label classification due to the fact the number of classes assigned to a document varies between different documents in the OHSUMED dataset (see Figure 2(b)). While pruning is based on one parameter, minimum confidence, ranking and selection can be performed in many ways. In particular three possibilities are considered (1) using confidence to prune rules, (2) using confidence to both prune and rank rules, and (3) using cosine of an angle between rules and documents to prune and rank rules. The following configurations are considered:

Simple. Rules are only pruned (based on minimum confidence). Ranking and selection are not performed in

this configuration.

Confidence factor. Rule ranking is based on rules' confidence. Selection is parametrized by *selection factor*, a value denoting the percentage of rules that should remain.

Cosine factor. Rule ranking is based on *cosine measure* between a rule and a document (represented by words). Cosine measure is a value equal to the angle between two vectors. Considering rule $\{o_1 i_1, o_2 i_2, ..., o_n i_n\} \rightarrow c$ and document $\{w_1 i_1, w_2 i_2, ..., w_n i_n\}$, where o_i and w_i are the numbers of occurrence of word i_i, the cosine measure is equal to $\arccos \angle(\vec{o}, \vec{w})$, where $\vec{o} = [o_1, o_2, ..., o_n]$ and $\vec{w} = [w_1, w_2, ..., w_n]$. Selection is performed using *selection factor* as in the previous configuration.

Simple, non-recurrent. Similar to the *Simple* configuration but rules are generated without considering the frequency of words in a document. Thus a rule is in the form of $\{i_1, i_2, ..., i_n\} \rightarrow c$.

Confidence factor, non-recurrent. The same as *Confidence factor* configuration but with non-recurrent item based rules.

Since non-recurrent item based classification does not carry information about the occurrences of words in a single document, we do not consider the *cosine factor* configuration in this case. Final set of classes is obtained by combining the classes that appear as a consequent in selected rules. We refer to the above configurations as R_{sim}, R_{con}, R_{cos}, S_{sim}, and S_{con}, respectively. To increase understanding of the above classification configurations, an example is shown in Figure 5.

document: $\{a, b, c, d, a, b, c, a, c, a, c\}$
recurrent item representation: $\{4a, 2b, 4c, 1d\}$
non-recurrent item representation: $\{a, b, c, d\}$

(a)

Rule #	Rule	confidence	cosine measure[1]
R_1	$\{3a, 2b\} \rightarrow C_1$	0.8	0.753
R_2	$\{2a, 1d\} \rightarrow C_1$	0.7	0.847
R_3	$\{1d, 2c\} \rightarrow C_2$	0.6	0.847
R_4	$\{2b, 1c\} \rightarrow C_3$	0.4	0.942

[1] Note that cosine measure is not an attribute of a rule and has to be calculated with respect to a given document

(b)

Config.	Min. conf.	Factor	Sel. rules	Assig. classes
R_{sim}	0.5	–	R_1, R_2, R_3	C_1, C_2
R_{con}	0.5	0.66	R_1, R_2	C_1
R_{cos}	0.5	0.66	R_1, R_2, R_3	C_1, C_2

(c)

Figure 5. Example of classification: (a) given document, (b) matching rules for *recurrent items* representation, and (c) classification results for different configurations

4. Experiments

4.1. Experimental setup

We used the OHSUMED collection [4] that contains of 348,543 records from the MEDLINE database limited to 5 years: 1987 to 1991. We reduced this collection to documents that have both a title and abstract which resulted in the total of 233,445 documents. This collection was divided into two subsets: (1) 183,229 documents dated 1987 through 1990 which were used 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], [19], [14]). However, unlike the others who used the testing set to tune the parameters of a classification process, we performed ten-fold cross validation on the training set, and used the obtained parameters (i.e. minimum confidence for all five configurations and confidence/cosine factor for R_{con}, R_{cos}, and S_{con}) to validate the classification system with the testing set. This approach is more strict and results in possibly lower performance, describing true non-overfitted models, when compared with other researchers. It is similar to [7] except that they computed the parameters by dividing the training set into only two subsets.
The generalization resulted in the total of 114 categories distributed as shown in Figure 2(b). Searching through the space of classification parameters took around 170 ten-fold cross validation experiments for all five configurations.

4.2. Quality evaluation

Evaluation of quality is based on commonly used measures, such as *accuracy*, *precision*, *recall*, and F_1 which combines the latter two [18]. The measures are based on a *contingency matrix* representing the number of *true positive* TP, *true negative* TN, *false positive* FP, and *false negative* FN classified examples. In multi-class classification there is also necessity of averaging single results from contingency matrices built for each class. We report two types of averaging: *macro averaging* and *micro averaging* [15]. Macro average, which is an arithmetical average of measures calculated for each class individually, emphasizes the ability of a classification system to behave well on all categories, even those with a low number of examples. Micro average, which is the average calculated by combining TP, TN,

Table 1. Training results

Consid. measure	Configuration	Min. conf.	Factor	Macro average					Micro average					Avg. F1	No. rules	Time [s]
				F1	σ_{F1}	Accur.	Prec.	Recall	F1	σ_{F1}	Accur.	Prec.	Recall			
Macro F1	R_{con}	0.3	0.4	**0.450**	0.0092	0.907	0.442	0.561	0.558	0.0085	0.907	0.468	0.692	0.504	22103	637
	R_{cos}	0.4	0.7	**0.449**	0.0105	0.894	0.434	0.585	0.543	0.0103	0.894	0.428	0.743	0.496	18562	446
	R_{sim}	0.5	–	**0.446**	0.0085	0.904	0.426	0.573	0.555	0.0070	0.904	0.458	0.705	0.501	15664	301
	S_{con}	0.3	0.7	**0.410**	0.0095	0.885	0.392	0.545	0.520	0.0095	0.885	0.402	0.737	0.465	13437	335
	S_{sim}	0.4	–	**0.406**	0.0097	0.893	0.404	0.520	0.529	0.0093	0.893	0.420	0.713	0.467	10742	216
Micro F1	R_{cos}	0.6	0.9	0.429	0.0117	0.927	0.540	0.443	**0.572**	0.0101	0.927	0.569	0.575	0.500	12906	271
	R_{sim}	0.6	–	0.437	0.0102	0.923	0.494	0.482	**0.571**	0.0085	0.923	0.538	0.608	0.504	12906	247
	R_{con}	0.5	0.6	0.437	0.0110	0.924	0.497	0.475	**0.569**	0.0090	0.924	0.544	0.598	0.503	15664	357
	S_{con}	0.4	0.6	0.385	0.0090	0.920	0.459	0.414	**0.551**	0.0096	0.920	0.522	0.584	0.468	10742	240
	S_{sim}	0.5	–	0.382	0.0094	0.919	0.436	0.415	**0.549**	0.0094	0.919	0.521	0.579	0.465	8554	170
Avg. F1	R_{cos}	0.5	0.8	0.446	0.0106	0.917	0.490	0.511	0.570	0.0091	0.917	0.507	0.651	**0.508**	15664	349
	R_{con}	0.4	0.5	0.447	0.0105	0.916	0.467	0.523	0.567	0.0086	0.916	0.503	0.649	**0.507**	18562	465
	R_{sim}	0.6	–	0.437	0.0102	0.923	0.494	0.482	0.571	0.0085	0.923	0.538	0.608	**0.504**	12906	247
	S_{con}	0.3	0.5	0.406	0.0099	0.905	0.431	0.483	0.545	0.0098	0.905	0.458	0.671	**0.475**	13437	335
	S_{sim}	0.4	–	0.406	0.0097	0.893	0.404	0.520	0.529	0.0093	0.893	0.420	0.713	**0.467**	10742	216

Table 2. Testing results

Consid. measure	Configuration	Min. conf.	Factor	Macro average				Micro average				Avg. F1	No. rules	Time [s]
				F1	Accur.	Prec.	Recall	F1	Accur.	Prec.	Recall			
Macro F1	R_{con}	0.3	0.4	0.459	0.905	0.449	0.566	0.566	0.905	0.480	0.689	0.512	21808	1938
	R_{cos}	0.4	0.7	0.455	0.892	0.438	0.590	0.553	0.892	0.440	0.744	0.504	18260	1320
	R_{sim}	0.5	–	0.454	0.901	0.431	0.581	0.563	0.901	0.467	0.709	0.508	15396	864
	S_{con}	0.3	0.7	0.420	0.883	0.408	0.553	0.532	0.883	0.416	0.738	0.476	13298	988
	S_{sim}	0.4	–	0.417	0.891	0.419	0.529	0.539	0.891	0.434	0.713	0.478	10629	622
Micro F1	R_{cos}	0.6	0.9	0.436	0.924	0.541	0.451	0.577	0.924	0.578	0.577	0.507	12619	779
	R_{sim}	0.6	–	0.446	0.920	0.499	0.491	0.577	0.920	0.548	0.609	0.511	12619	700
	R_{con}	0.5	0.6	0.446	0.921	0.502	0.484	0.575	0.921	0.554	0.598	0.510	15396	1047
	S_{con}	0.4	0.6	0.395	0.917	0.465	0.425	0.560	0.917	0.535	0.588	0.478	10629	699
	S_{sim}	0.5	–	0.391	0.917	0.443	0.424	0.558	0.917	0.534	0.583	0.475	8431	485
Avg. F1	R_{cos}	0.5	0.8	0.452	0.914	0.494	0.519	0.578	0.914	0.516	0.656	0.515	15396	1020
	R_{con}	0.4	0.5	0.456	0.913	0.483	0.532	0.573	0.913	0.514	0.648	0.515	18260	1384
	R_{sim}	0.6	–	0.446	0.920	0.499	0.491	0.577	0.920	0.548	0.609	0.511	12619	700
	S_{con}	0.3	0.5	0.416	0.903	0.442	0.492	0.554	0.903	0.472	0.671	0.485	13298	987
	S_{sim}	0.4	–	0.417	0.891	0.419	0.529	0.539	0.891	0.434	0.713	0.478	10629	622

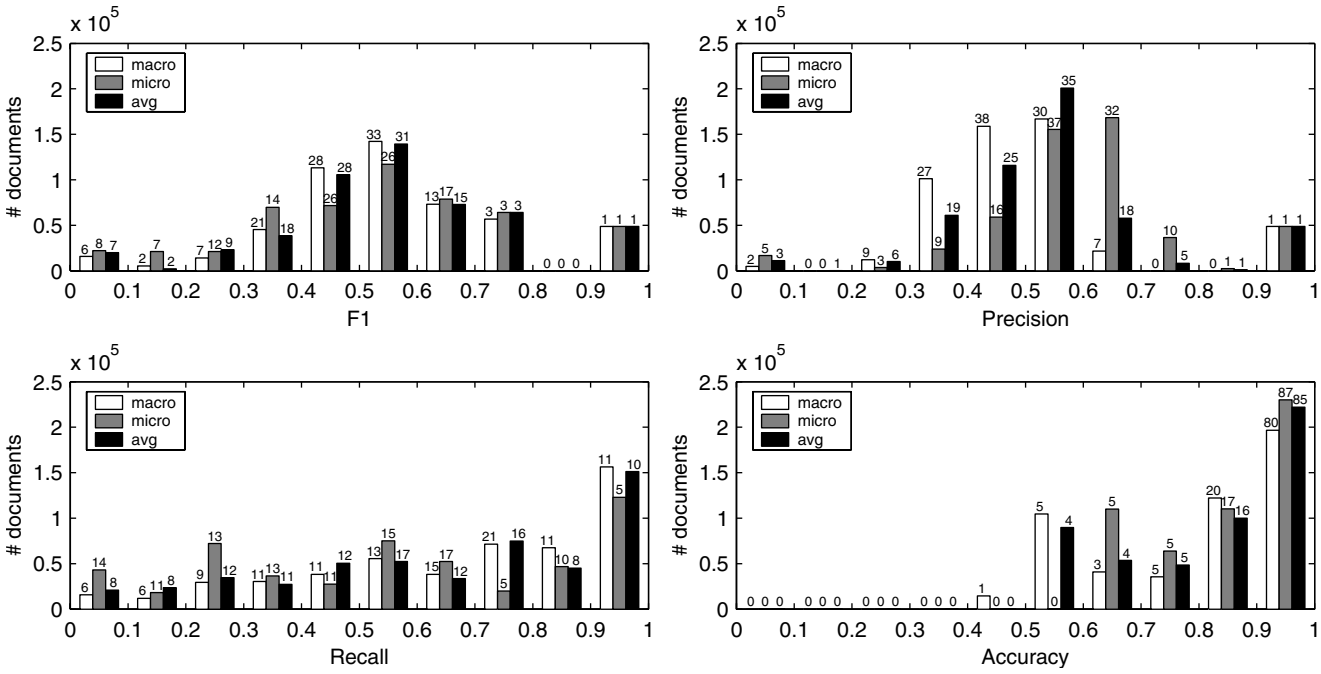
FP, and FN examples across all categories into one contingency matrix, reflects better classification for larger classes in expense of poorer results for small ones. We put special emphasis on F_1, the measure used by other researches, but report also precision, recall, and accuracy to give further insight.

4.3. Results

Experiments performed on the training set using 10-fold cross validation resulted in selection of three sets of parameters for each configuration. The three sets are selected to maximize the value of *macro* F_1, *micro* F_1, and the average of those two, to present a detailed comparison of differences between micro and macro averaging. All 15 sets of parameters together with quality measures are presented in Table 1. The best results for each considered measure are at the top of each group of rows in the table. In order to demonstrate the correctness of chosen parameters, experiments on the testing set using the same 15 sets of parameters were performed. The results of testing are shown in Table 2. For the testing data the order of the configurations with respect to considered measures is the same. Only one set of parameters is common for different measures (micro F_1 and average of micro and micro F_1 for R_{sim}). Thus the choice of the right configuration and parameters depends on the type of measure being considered. For macro averaging, the best results are obtained for R_{con}, whereas R_{cos} works better with micro averaging and in case of considering the average between macro and micro average. This is consistent for both cross-validation performed on training data and test ran on independent test set. Similarity in the results in both training and test data evaluations gives high confidence in subsequently described conclusions.

The results indisputably show that configurations utilizing the knowledge of occurrences of words in a document are better than those that neglect this information. A difference is especially visible in macro F_1 score and reaches about 0.04. The best result when recurrent items are used is 0.46, whearas the best result for non-recurrent items is 0.42. Similar difference is also observed for both micro F_1 and average F_1. On the other hand average classification time is lower for configurations with non-recurrent items. In case of macro F_1 the difference is two-fold, whereas for micro F_1 and average F_1 is less substential.

Figure 6 shows how the documents from the testing set are distributed with respect to F_1, precision, recall, and accuracy, for the three best configurations that correspond to mi-



Numbers above bars indicate a number of categories

Figure 6. Document distribution with respect to F_1, precision, recall, and accuracy

cro, macro, and average F_1. The distribution over F_1 score for the best micro average configuration is slightly flatter than this for the best macro average configuration. However, substantial differences in the distribution over precision and recall can be observed. Precision distribution shows that more documents are classified with higher precision for the best micro F_1 configuration, whereas more documents have better recall for the best macro average configuration. Precision is inversely proportional to FP examples and thus its low values show tendency for "overclassification", i.e. classification of a single document to additional incorrect categories, which is a result of too general models. On the other hand, recall is inversely proportional to FN examples, reflecting inability of assigning classes to a document, i.e. the model is too specific. Observation based on Figure 6 go in line with the nature of macro and micro average. Micro average is in essence weighted average and thus is more suitable when a user is interested in maximizing performance of categories with large number of examples, neglecting, to a certain extent, the classification quality of categories with relatively low number of examples. That is why in case of maximizing micro F_1 more documents have higher precision when compared with maximization of macro F_1.

Based on experiments a user has the following choices:

1. When parameters corresponding to maximization 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 (classes), some of the categories may not be found.

2. When parameters corresponding to maximization of macro F_1 are used, higher recall and lower precision will be achieved. In this case less categories will be omitted, but among the classification results more will be incorrect.

3. Using average of macro an micro F_1 gives the trade-off between the two above situation.

It is left to a user to decide which option to choose, i.e. to make more mistakes, but have more categories assigned to documents, to make less mistakes, but have less categories assigned, or finally to use a setting that results in something in the middle.

To the best of our knowledge no-one has tried to generalized the MeSH tree in a manner described in this paper. Only 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 of Lam et al. [7] who used instance-based learning and retrieval feedback. They obtained macro F_1 score of 0.44 using MeSH categories with at least 75 examples (documents) in the training set.

5. Conclusions

This paper describes the development of the system for classification of medical article references. The proposed system is based on associative classification technology. We used OHSUMED, a subset of the MEDLINE database, as the source of documents and the MeSH tree as class labels. The specific feature of classification of medical documents to MEDLINE is that each document is assigned to multiple categories, i.e. multi-label classification must be performed. The second major challenge was to develop a scalable method capable of dealing with hundreds of thousand of documents, which is characteristic to this medical repository. We employed the recently developed ACRI tool, which was modified to accommodate for multi-label classification. Five different classification configurations in conjunction with different methods of measuring classification quality were used to perform classification. The extensive experimental comparison shows superiority of recurrent item based methods vs. non-recurrent based associative classification. High quality of the developed system is justified by relatively high value of 0.46 of macro F_1 and over 90% accuracy. Additionaly, three conifurations were proposed. If the goal is to classify the largest number of documents, one should choose a configuration that maximizes micro F_1. On the other hand, if one wishes our system to work well for categories with small number of documents a configuration that maximizes macro F_1 should be chosen. A trade-off is obtained by using a configuration that optimizes the average between macro and micro F_1.

References

[1] R. Agarwal and R. Srikant. Fast algorithms for mining association rules. In *International Conference Very Large Data Bases*, pages 487–499, 1994.

[2] M.-L. Antonie, O. R. Zaïane, and A. Coman. Associative classifiers for medical images. *Lecture Notes in Computer Science*, 2797:68–83, 2003.

[3] C. Blake and W. Pratt. Better rules, few features: A semantic approach to selecting features from text. In *Proceedings IEEE International Conference on Data Mining*, pages 59–66, 2001.

[4] W. Hersh, C. Buckley, T. J. Leone, and D. Hickam. Ohsumed: an interactive retrieval evaluation and new large test collection for research. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 192–201, 1994.

[5] P. Kankar, S. Adak, A. Sarkar, K. Murali, and G. Sharma. MedMeSH summarizer: Text mining for gene clusters. In *Proceeding of the 2nd SIAM International Conference on Data Mining*, pages 548–565, 2002.

[6] W. Lam and C. Y. Ho. Using a generalized instance set for automatic text categorization. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 81–89, 1998.

[7] W. Lam, M. Ruiz, and P. Srinivasan. Automatic text categorization and its application to text retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 11(6):865–879, 1999.

[8] D. D. Lewis, R. E. Schapire, J. P. Callan, and R. Papka. Training algorithms for linear text classifiers. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 298–306, 1996.

[9] W. Li, J. Han, and J. Pei. CMAR: Accurate and efficient classification based on multiple class-association rules. In *Proceedings IEEE International Conference on Data Mining*, pages 369–376, 2001.

[10] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. In *Knowledge Discovery and Data Mining*, pages 80–86, 1998.

[11] National Medicine Library. MeSH's fact sheet, http://www.nlm.nih.gov/pubs/factsheets/mesh.html, 2005.

[12] M. F. Porter. An algorithm for suffix stripping. *Readings in information retrieval*, pages 313–316, 1997.

[13] R. Rak, W. Stach, O. R. Zaïane, and M.-L. Antonie. Considering re-occurring features in associative classifiers. *Lecture Notes in Computer Science*, 3518:240–248, 2005.

[14] M. E. Ruiz and P. Srinivasan. Hierarchical text categorization using neural networks. *Information Retrieval*, 5(1):87–118, Jan 2002.

[15] F. Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, 2002.

[16] F. Thabtah, P. Cowling, and Y. Peng. MMAC: a new multi-class, multi-label associative classification approach. In *Proceedings of Fourth IEEE International Conference on Data Mining, 2004. ICDM 2004*, pages 217–224, 2004.

[17] N. Uramoto, H. Matsuzawa, T. Nagano, A. Murakami, H. Takeuchi, and K. Takeda. A text-mining system for knowledge discovery from biomedical documents. *IBM Systems Journal*, 43(3):516–533, 2004.

[18] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition, 1979.

[19] Y. Yang. An evaluation of statistical approaches to MEDLINE indexing. In *Proceedings of the Conference of the American Medical Informatics Association*, pages 358–362, Washington, D.C., 1996.

[20] X. Yin and J. Han. CPAR: Classification based on predictive association rules. In *Proceedings of 3rd SIAM International Conference on Data Mining (SDM'03)*, 2003.

[21] O. Zaïane, J. Han, and H. Zhu. Mining recurrent items in multimedia with progressive resolution refinement. In *Proceedings of 16th International Conference on Data Engineering*, pages 461–470, 2000.

[22] O. R. Zaïane and M.-L. Antonie. Classifying text documents by associating terms with text categories. In *Thirteenth Australasian Database Conference*, pages 215–222, 2002.