New Learning Strategies for Automatic Text Categorization

賴 國 賢
LAI Kwok-yin

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

© The Chinese University of Hong Kong
June 2001

摘要

隨著國際互聯網絡應用日漸普及，文字檔案的數目與日俱增，要從數目龐大的文件集中，檢索某一文件檔案，是一件需時費力的工作，自動化文件歸類是現今的主要技術，幫助我們整理大量的文件集。

在這篇論文中，我們研究了新的總學習模式應用在自動化文件歸類上。 總學習的主旨是合併現存各種分類法的優點進行歸類工作，從而改善分類表現 ，我們就此模式提出了三種新方法，它們分別是線性結合法，文件特性 總學習法，以及兩者結合之新方法。

線性結合法的特質是可替我們對各種分類方法在不同類別的相對表現作 估計而進行演繹。在線性結合法的架構下，我們提出了三種不同的策略， 這些策略爲決定各種分類方法在最終的分類判斷中所佔的相對重要性提供 了一套準則。

我們所提出的第二種總學習法，是結合了多元回歸分析法及，收集各類別獨有 的文件特性。通過學習類別獨有的文件特性以及各分類法的分類誤差兩者之 間的關係，我們可預測各種分類法的表現，基於獲得的預測表現，這新方 法便可爲每一個文件類別建議採用指定的分類法進行文件歸類。

通過結合線性結合法及文件特性總學習法，我們進一步提出第三種嶄新的 總學習歸類法。這方法可以在爲每一種分類法決定相度比重因

數的同時 ，更可考慮類別獨有的文件特性。決定比重因數而收集文件特性 的好處，是可對各文件類別特質有了綜合性了解，從而使各分類方法的 相對重要性更眞實地反映出來。

我們利用了兩個現實大文件集進行了實驗。實驗結果顯示我們所提出的總 學習歸類法，在不同方向的比較下，較其他現存的單一分類法更優勝。

Abstract

With the increasing use of the Internet, the volume of textual document collection is also increasing rapidly. Given a huge volume of text data, it certainly takes considerable amount of time and effort to retrieve a piece of document from the collection. Automatic text categorization is a major technique to organize a large document collection.

In this thesis, instead of making refinement for particular classification algorithms, we conduct research on new approaches for meta-learning models of automatic textual document categorization. Meta-learning techniques aim to combine and unify the strength of existing component classification algorithms in order to obtain an improved overall classification performance. We have investigated three meta-learning approaches for automatic text categorization, namely, the Linear Combination approach (LC), the Meta-learning Using Document Feature characteristics (MUDOF), and MUDOF2.

The Linear Combination approach can distill the characteristics of how we estimate the relative merit of each component algorithm for different categories. Under the linear combination framework, LC, we propose three different strategies, which are used for determining the relative contribution of the component algorithms towards the final classification decisions.

We propose our second new meta-learning approach, MUDOF, for text categorization, based on multivariate regression analysis, by capturing category specific document feature characteristics. By learning the relationship between categorical feature characteristics and the classification errors of different algorithms, classification performance of each component algorithm are predicted. Based on the predicted performance, the approach is able to recommend the most ideal component algorithms for each category.

By incorporating MUDOF into Linear Combination framework, we further propose the third meta-learning approach, MUDOF2. MUDOF2 can derive the relative weight factors for each component classification algorithm with proper consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category.

Extensive experiments have been conducted on two large-scale, real-world document collections, namely, the Reuters-21578 and the OHSUMED corpus. Results show that our proposed approaches demonstrate overall better classification performance over individual component algorithms under different perspectives of evaluation.

Acknowledgments

I would like to express my sincere gratitude to my research advisor, Prof. Wai Lam. His invaluable advice and continuous inspiration have contributed a great deal to this research work.

I would also like to thank Prof. Christopher Yang and Prof. Jeffrey Yu for their opinions on improving the quality of my work.

Furthermore, I feel extremely grateful to my family and Katherine for their tremendous care and continuous support. Their encouragement and understanding helped me to progress and concentrate on my work.

Last but not least, I would like to thank my friends in the Department of Systems Engineering and Engineering Management of the Chinese University of Hong Kong (CUHK). I want to mention particularly Lin Wai Yip, Sally Yau, Connie Tsui, Philip Lee, Su Yat Fan and Lee Sung Tak. Their encouragement and help made my life in CUHK delightful and enjoyable.

Lai Kwok-Yin, Patrick.

June 2001.

Contents

1 Introduction 1

1.1 Automatic Textual Document Categorization 1

1.2 Meta-Learning Approach For Text Categorization 3

1.3 Contributions 6

1.4 Organization of the Thesis 7

2 Related Work 9

2.1 Existing Automatic Document Categorization Approaches . . 9

2.2 Existing Meta-Learning Approaches For Information Retrieval 14

2.3 Our Meta-Learning Approaches 20

3 Document Pre-Processing 22

3.1 Document Representation 22

3.2 Classification Scheme Learning Strategy 25

4 Linear Combination Approach 30

4.1 Overview 30

4.2 Linear Combination Approach - The Algorithm 33

4.2.1 Equal Weighting Strategy 34

4.2.2 Weighting Strategy Based On Utility Measure 34

4.2.3 Weighting Strategy Based On Document Rank 35

4.3 Comparisons of Linear Combination Approach and Existing Meta-Learning Methods 36

4.3.1 LC versus Simple Majority Voting 36

4.3.2 LC versus BORG 38

4.3.3 LC versus Restricted Linear Combination Method . . . 38

5 The New Meta-Learning Model - MUDOF 40

5.1 Overview 41

5.2 Document Feature Characteristics 42

5.3 Classification Errors 44

5.4 Linear Regression Model 45

5.5 The MUDOF Algorithm 47

6 Incorporating MUDOF into Linear Combination approach 52

6.1 Background 52

6.2 Overview of MUDOF2 54

6.3 Major Components of the MUDOF2 57

6.4 The MUDOF2 Algorithm 59

7 Experimental Setup 66

7.1 Document Collection 66

7.2 Evaluation Metric 68

7.3 Component Classification Algorithms 71

7.4 Categorical Document Feature Characteristics for MUDOF and MUDOF2 72

8 Experimental Results and Analysis 74

8.1 Performance of Linear Combination Approach 74

8.2 Performance of the MUDOF Approach 78

8.3 Performance of MUDOF2 Approach 87

9 Conclusions and Future Work 96

9.1 Conclusions 96

9.2 Future Work 98

A Details of Experimental Results for Reuters-21578 corpus 99

B Details of Experimental Results for OHSUMED corpus 114

Bibliography **125**

List of Figures

1.1 The framework of a generic automatic document categorization system . . . 3

1.2 The meta-learning framework for automatic document categorization system . . . 5

3.1 Internal representation of training documents collection 25

3.2 An example of internal representation of training documents collection . . . 27

3.3 A sample document in the Reuters-21578 document collection 29

4.1 The framework of applying linear combination technique for a particular category . . . 32

5.1 An overview of the meta-model approach for text categorization 43

5.2 The MUDOF algorithm . . . 51

6.1 An overview of MUDOF2 . . . 55

List of Tables

6.1 The MUDOF2 algorithm 63

8.1 Comparison of Linear Combination approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus. 76

8.2 Comparison of classification performance of Linear Combination approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus. 77

8.3 Percentage improvement of the Linear Combination approach over existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus. . . 77

8.4 Comparison of Linear Combination approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus. 79

8.5 Comparison of classification performance of Linear Combination approach with existing component classification algorithms in different perspectives of measure for the OHSUMED corpus. 80

8.6 Percentage improvement of the Linear Combination approach over existing component classification algorithms under different perspectives of measure for the OHSUMED corpus. . . . 80

8.7 Parameter estimates for categorical document feature characteristics of different algorithms for the Reuters-21578 corpus. 83

8.8 Comparison of the MUDOF approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus. 84

8.9 Comparison of classification performance of the MUDOF approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus. 85

8.10 Improvement of classification performances of MUDOF (M+(%)) and the ideal combination (I+(%)) over individual classification algorithms, and improvement achieved by MUDOF within the improvement bound set by the ideal combination (M+/I+(%)) for the Reuters-21578 corpus. 85

8.11 Parameter estimates for categorical document feature characteristics of different algorithms for the OHSUMED corpus. . . 87

8.12 Comparison of the MUDOF approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus. 88

8.13 Comparison of classification performance of the MUDOF approach with existing component classification algorithms under different perspectives of measure for the OHSUMED corpus. 89

8.14 Improvement of classification performances of MUDOF (M+(%)) and the ideal combination (I+(%)) over individual classification algorithms, and improvement achieved by MUDOF within the improvement bound set by the ideal combination (M+/I+(%)) for the OHSUMED corpus. 89

8.15 Comparison of the MUDOF2 approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus. 91

8.16 Comparison of classification performance of the MUDOF2 approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus. 92

8.17 Improvement of classification performances of MUDOF2 (M2+(%)) and the Linear Combination approach with Weighting Strategy based on Utility Measure (LC2+(%)) over individual classification algorithms for the Reuters-21578 corpus. 92

8.18 Comparison of the MUDOF2 approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus. 94

8.19 Comparison of classification performance of the MUDOF2 approach with existing component classification algorithms under different perspectives of measure for the OHSUMED corpus. 95

8.20 Improvement of classification performances of MUDOF2 (M2+(%)) and the Linear Combination approach with Weighting Strategy based on Utility Measure (LC2+(%)) over individual classification algorithms for the OHSUMED corpus. 95

A.1 Complete comparison of Linear Combination approach under the Equal Weighting Strategy (LC1) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus. 101

A.2 Complete comparison of Linear Combination approach under the Weighting Strategy Based On Utility Measure (LC2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus. 104

A.3 Complete comparison of Linear Combination approach under the Weighting Strategy Based On Document Rank (LC3) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus. 107

A.4 Complete comparison of MUDOF approach (MUDOF) and the ideal combination of algorithms (IDEAL) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus. 110

A.5 Complete comparison of MUDOF2 approach (MUDOF2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus. 113

B.1 Complete comparison of Linear Combination approach under the Equal Weighting Strategy (LC1) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus. 116

B.2 Complete comparison of Linear Combination approach under the Weighting Strategy Based On Utility Measure (LC2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus. 118

B.3 Complete comparison of Linear Combination approach under the Weighting Strategy Based On Document Rank (LC3) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus. 120

B.4 Complete comparison of MUDOF approach (MUDOF) and the ideal combination of algorithms (IDEAL) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus. 122

B.5 Complete comparison of MUDOF2 approach (MUDOF2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus. 124

Chapter 1

Introduction

In this chapter, we first give the problem definitions of automatic textual document categorization. Then, we present the motivation of our meta-learning approach for the task. We summarize our major studies and contributions of our research. Finally, the organization of this thesis will be outlined.

1.1 Automatic Textual Document Categorization

Electronic mails, news articles, books, journals or technical manuals, are all examples of textual documents, or text data, that we discuss in our work. With the increasing use of the Internet, the volume of textual document collection is also increasing rapidly. Given a huge volume of text data, it certainly takes considerable amount of time and effort to retrieve a piece of document from the collection.

One way to organize a large document collection is to conduct *docu-*

ment categorization. Typically, there is a set of category labels which are pre-defined in advance. The task of document categorization is to assign a number of appropriate categories to each textual document. To conduct document categorization, the full context of the textual documents have to be understood before assigning appropriate categories.

Traditionally, this task is performed manually by domain experts. Each incoming document has to be analyzed by the expert based on the content of the document. However, performing manual document categorization obviously involves a great deal of time and human effort. As a result, the aim of *automatic textual document categorization* is to classify textual documents into appropriate categories automatically.

The goal of automatic text categorization is to construct a classification scheme, or called a classifier, from the training data set by means of machine learning. A classifier captures the context and nature for a category by using a training data set. A training data set contains sample documents and their corresponding categories. Specifically, there is a classification scheme learned for each category during the training phase. After completing the whole training phase, each category is associated with a different classification scheme which is used for categorizing future documents automatically. Figure 1.1 depicts the major tasks involved in an automatic textual document categorization system. The purpose of the Document Pre-processing Task is to convert a document into an internal representation which can be processed in the system. The purpose of the Classification Learning Task is to learn a classification scheme from training documents. Each category is associated with one classification scheme. The purpose of the On-line Classification Task is to decide the category membership for the new documents based on the learned classification schemes.

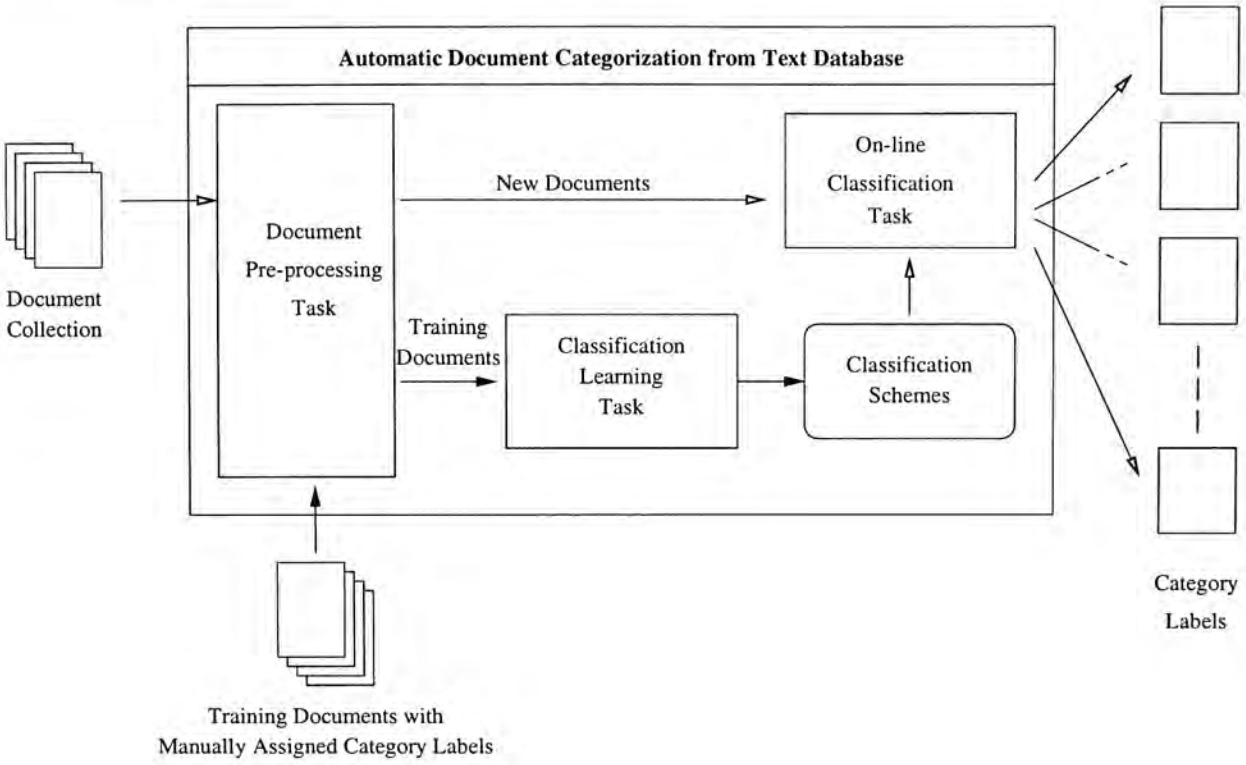
Automatic Document Categorization from Text Database
Document Collection
Document Pre-processing Task
New Documents
On-line Classification Task
Training Documents
Classification Learning Task
Classification Schemes
Category Labels
Training Documents with Manually Assigned Category Labels

Figure 1.1: The framework of a generic automatic document categorization system

1.2 Meta-Learning Approach For Text Categorization

A variety of classification algorithms have been proposed in the information retrieval (IR) community. Most of the newly refined and proposed approaches are reported to demonstrate classification improvement over existing algorithms. Yet, they are developed based on a single paradigm to solve the categorization problem. Indeed, some recent work [22, 20, 19, 25, 3, 2] suggesting that further improvements in classification performance can be achieved by combining multiple evidence from more then one classification algorithms. Such method of combining various classification models to tackle the classification problem is collectively called *meta-learning approach*.

Instead of making refinement to particular classification algorithms, the aim of meta-learning approach for text categorization is to unify and combine the strength of different algorithms in order to achieve an overall better classification performances over individual classification models, or *component classifiers*. Studies of meta-learning techniques in the IR literature is not uncommon, but those of applying the technique specifically on text categorization are all based on simple linear combination of several basic algorithms.

The general framework of meta-learning approach for text categorization is similar to the traditional text categorization algorithm as described previously, except the classification scheme to be constructed during the training phase. Instead of using only one algorithm, meta-model learning involves more than one categorization algorithm. Under the approach, classification schemes that have been separately learned by different algorithms for a category, are combined together in a certain way, to yield one single meta-model classification scheme. Given a document to be categorized, the meta-model classification scheme can be used for deciding the document membership for the category. As a result, each meta-model classifier for a category is the combined contributions of all the involved algorithms. Figure 1.2 depicts the major tasks involved in an automatic textual document categorization system using meta-learning technique. Notice that the Document Pre-processing

Task and is the same as that of generic automatic document categorization system as depicted in Figure 1.1. However, instead of involving only one single Classification Learning Task, employing meta-learning approach involves several Classification Learning Tasks. Each of them is performed by a particular classification algorithm. The component classifiers are combined to yield the Meta-Classification Schemes, which are used for classifying the new documents. The On-line Classification Task is also designed to be fitted into the meta-learning framework.

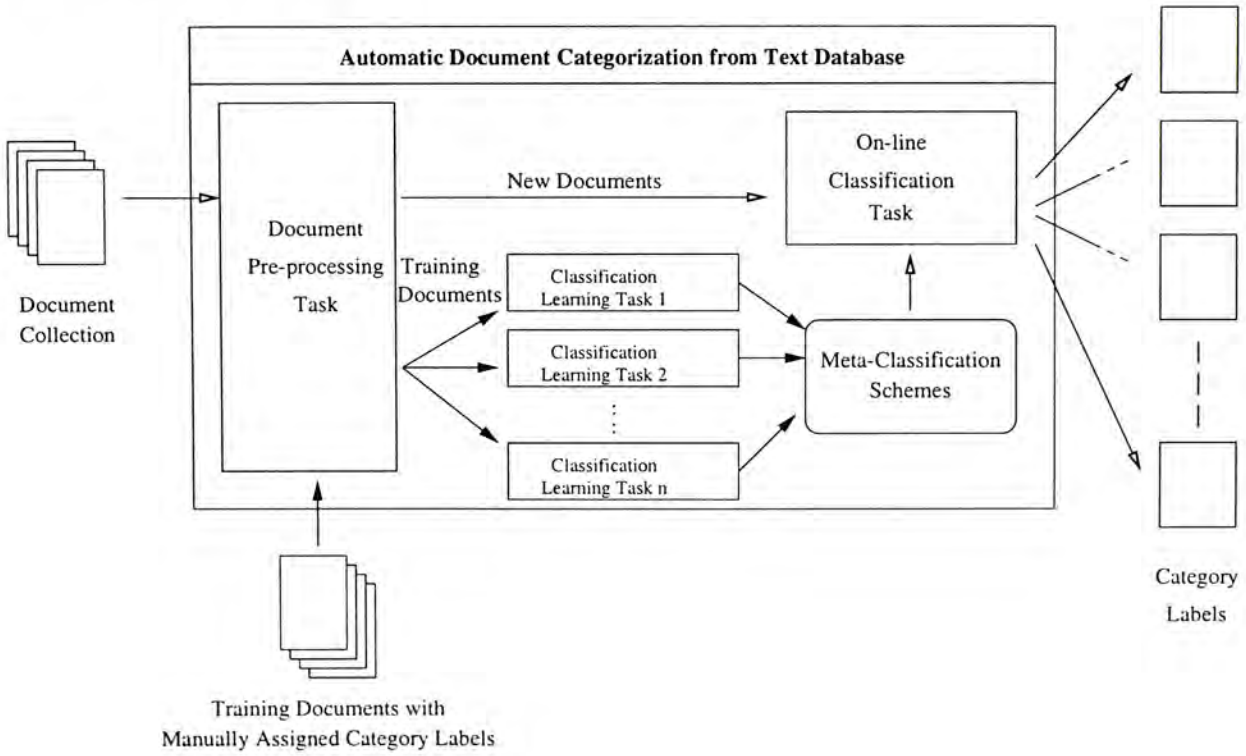
Automatic Document Categorization from Text Database
Document Collection
Document Pre-processing Task
New Documents
On-line Classification Task
Training Documents
Classification Learning Task 1
Classification Learning Task 2
Classification Learning Task n
Meta-Classification Schemes
Category Labels
Training Documents with Manually Assigned Category Labels

Figure 1.2: The meta-learning framework for automatic document categorization system

1.3 Contributions

We have conducted research on new approaches for meta-learning models of automatic textual document categorization. We investigate the Linear Combination approach (LC) by distilling the characteristic of how we estimate the relative merit of each component algorithm for different categories. Based on this idea, we propose three different strategies for the Linear Combination approach. The approach makes use of limited knowledge in the training document set. To address this limitation, we propose a second meta-learning approach, called Meta-learning Using Document Feature characteristics (MUDOF). By incorporating MUDOF into Linear Combination framework, we further propose MUDOF2 to tackle the text categorization problem. The major contributions are summarized as follows:

- The linear combination approach can distill the characteristics of how we estimate the relative merit of each component algorithm for different categories. Under the linear combination framework, LC, we propose three different weighting strategies, which are used for determining the relative contribution of the component algorithms towards the final classification decisions.

- We propose our second meta-learning approach, MUDOF, for text categorization, based on multivariate regression analysis, by capturing category specific feature characteristics. By learning the relationship between categorical document feature characteristics and the classification errors of different algorithms, the approach can predict the classification performance of each component algorithm. Based on the predicted errors, the approach is able to recommend the most ideal component algorithms for each category. Experimental results confirm that capturing categorical document feature characteristics helps to improve the overall classification performances.

- By combining both LC and MUDOF approaches, we further propose the third meta-learning approach, MUDOF2. Different from the Linear Combination approach, MUDOF2 can derive the relative weight factors for each component classification algorithm with consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category. Experimental results show that MUDOF2 can not only improve the classification performances of the component algorithms, but also largely improve the Linear Combination under the Weighting Strategy Based On Utility Measure.

1.4 Organization of the Thesis

The thesis starts to present a survey on the existing approaches for automatic document categorization and several representative meta-learning algorithms in the literature in Chapter 2. The pre-processing of textual documents into internal representation is described in Chapter 3. In Chapter 4, a generalized version of Linear Combination approach is presented in details. Chapter 5 describes the new meta-learning approach called MUDOF. By incorporating MUDOF into the Linear Combination approach, we present another new meta-learning approach called the MUDOF2 in Chapter 6. Chapter 7 describes the experimental setup for the evaluation of our proposed approaches. Chapter 8 shows the experimental results after our extensive runs of experiments with our analysis. Chapter 9 gives the conclusions and the future work.

Chapter 2

Related Work

In this chapter, we present the representative related work of automatic textual document categorization and also meta-learning approaches.

2.1 Existing Automatic Document Categorization Approaches

Rule-based learning approaches are one of the early techniques that are considered to have good performance to tackle the problem of text categorization.

An early work performed by Apte et al. [1] adopted a decision tree learning technique to learn a classifier in the form of a decision tree. By using optimized rule-based induction methods, the technique can derive category assignment rules automatically from samples of documents to be classified. The identified classification patterns are applied on text categorization.

Cohen and Singer [5] employed the RIPPER approach, a rule learning algorithm, and developed the sleeping experts algorithm which is based on a multiplicative weight update technique for text categorization. Both algorithms allow the context of a word to influence how the presence or absence of that word will affect a classification decision. Their work confirms that building classifiers that capture contextual information can increase general classification performance.

Tan [40] has recently introduced the use of predictive self-organizing neural networks for classification of textual documents. The approach incorporates the rule-insertion mechanism which can integrate the domain specific knowledge into the on-line classification task in order to achieve improved classification performance.

Koller and Sahami [17] explored hierarchical categorization of documents by combining probabilistic framework and reduction of feature space. Mainly, the approach breaks the classification problem into a set of smaller classification tasks. At each decision point of the hierarchy, a classifier is constructed by a learning algorithm. Each of the classifiers are constructed based on their own set of relevant features.

Similar work has been performed by Wang et al. [43], who attempted to build hierarchical classifiers using class proximity. By considering the closeness of features towards a target class and by constructing a global classifier carrying global information across classes, good classification performance is observed.

Yang and Chute [47] proposed a statistical approach known as Linear Least Squares Fit (LLSF) which estimates the likelihood of the associations between document terms and categories via a linear parametric model.

Inference-based Bayesian approach, a widely studied and consistently showing good classification performance, is another popular learning algorithm that is applied on text categorization.

Lam et al. [23] attempts tackle this problem using Bayesian network approach, which is an improved approach over the Bayesian independence classifiers. The use of Bayesian network can eliminate the assumption that the feature terms are independent to a category and a document.

Meretakis et al. [31] examined text categorization methods, including support vector machines (SVM) and Bayesian extensions of Naïve Bayes Classifier, and compared the tradeoff between accuracy, scalability to large data set and large feature sizes. The study shows that the Bayesian extension to Naïve Bayes can achieve good tradeoff between high classification accuracy and scalability to large document collections and large feature sizes.

McCallum and Nigam [29] investigated and compared the classification performances of two models of Naïve Bayes approach for text categorization. Empirical results show that the multinomial model uniformly performs better than the multi-variate Bernoulli event model. The findings confirm that capturing word frequency information in document can achieve more satisfactory classification performance with traditional Bayesian network approach.

Recently, Nigam et al. [33] introduced a classification algorithm for learning from both labeled and unlabeled documents based on the combination of Naïve Bayes classifier and iterative algorithms for maximum likelihood estimation with incomplete data. The study demonstrates that unlabeled documents do contribute to the overall better classification performance.

Linear classifiers are another learning technique that demonstrate good classification performance and ease of implementation.

Lewis [27] explored two linear classifiers, namely the Widrow-Hoff (WH) algorithm and the exponentiated-gradient (EG) algorithm proposed by Kivinen and Warmuth, for the text categorization problem. Binary decision on the relevance of a document against a category is made by considering the similarity between the document's feature vector and the classifier, or the weight vector, of the category, with respect to a threshold. Results show that both WH and EG algorithms demonstrates better classification performance than Rocchio.

The K-Nearest Neighbor (KNN) approach is a well-known technique that is simple, yet achieve very good classification performance over some traditional classification algorithm. There have been a lot of published studies reporting the robustness of KNN, and a lot of improved variants have been proposed recently.

Yang [44] developed an algorithm known as ExpNet which derives from the k-nearest neighbor technique, an instance-based classification approach. ExpNet achieves good categorization performance on large document corpora such as the Reuters collection and the OHSUMED collection.

In a recent work, Yang [46] further proposed several improved variants of original KNN classification algorithm for event tracking. The new variants of KNN can successfully reduce the error rate significantly on several topic detection and tracking document collections.

Han [10] also proposed the weight adjusted KNN classification method, a new KNN approach based on a greedy hill climbing technique. By combining two performance optimization techniques, the new approach outperforms

C4.5, RIPPER and Naïve Bayes in computational time and classification performance.

Instead of constructing one single classifier for each category, Lam and Ho [21] propose the use of generalized instance set (GIS) approach to construct a set of generalized instances for each category. By combining the technique of KNN and linear classifiers, GIS shows better classification performance over existing KNN and two linear classifiers, namely the WH and Rocchio algorithm.

Support vector machines (SVM) is a relatively new learning algorithm proposed by Vapnic [42]. The approach is based on the Structural Risk Minimization principle. SVM aims to learn a hyperplane which can linearly separate documents into either the class of belonging to a category or the class that does not belong to the category. In various recent studies, SVM demonstrate a better performance over other existing text categorization algorithms.

Joachims [14], as well as, Yang and Liu [48] compared SVM with KNN and Naïve Bayes classifier. Both results show that SVM outperforms both KNN and Naïve Bayes classifier.

Dumais et al. [7] compared SVM, decision trees, Bayesian network and Naïve Bayes approaches on the Reuters collection. Results show that not just SVM's classification performance is better than other methods.

Karypis and Han [15] explored a method of supervised dimensionality reduction algorithm, which leads to increased document categorization and retrieval performance over C4.5, KNN and SVM. Adding their proposed concept indexing approach, SVM can outperform KNN on a Reuters corpus.

Application of SVM on classifying email as spam or nonspam has also been studied by Drucker et al. [6]. The study shows that SVM can have good performance in terms of accuracy and shorter training and classification time when compared with Rocchio linear classifier.

2.2 Existing Meta-Learning Approaches For Information Retrieval

In machine learning community, a lot of different methods on meta-learning, or multi-strategy learning, have been proposed.

Boosting method, one of the meta-learning strategies proposed recently, combines the weak hypotheses, which are sequentially learned by the same learning method, called the weak learner. At each iteration of a boosting method, a weak hypothesis is learned by taking into account how the weak hypotheses, that are learned in the previous iterations, perform on the training documents. Weights will get incrementally higher for those document examples that are hard to learn. The weak learner will concentrate on those documents due to the increased weight in the succeeding iterations, which helps to find the final highly accurate classification rule. After a specific number of iterations, a final hypothesis is obtained by combining all the weak hypotheses. The final hypothesis is then used to classify the unseen documents.

AdaBoost, a commonly studied boosting algorithm, is proposed by Freund and Schapire [9] to solve binary classification problems. Two variants are proposed based on AdaBoost to make it able to handle classification problems. The work shows that there exists an error bound for the algorithms, which guarantees its classification performance. Theoretical proof has been done in the study. However, empirical evidence of the algorithms on text categorization is not given.

Based on AdaBoost, Schapire and Singer [37] proposed a new family of boosting algorithms for text categorization. Their work shows that their variants of AdaBoost achieve very satisfactory classification performances over Rocchio and Naïve Bayes.

Sebastiani et al. [38] recently proposed an improved boosting algorithm based on AdaBoost for text categorization by generating a set of, rather than only one single, weak hypotheses at each iteration of the boosting process. The set of hypotheses are combined by simple arithmetic mean to produce the weak hypothesis for that iteration. Results are compared with the variants proposed by Schapire and Singer [37], showing that the new method demonstrate a better classification performances.

Iyer et al. [13] investigated the behavior of RankBoost [8] on different ranking functions for the weak hypotheses in the context of document routing.

Boosting algorithms combine weak hypothesis into one single final hypothesis by using a weak learner. This combination of evidence obtained by multiple runs is based on a single learning method. Evidence combined is generated by the same algorithm of the same type.

Recently, there have been many studies which relax the restriction of combining evidence generated by one single classification method. Instead of considering only one algorithm's evidence, several meta-learning work have proposed to combine the evidence of predictions learned by algorithms, which are of different nature and types.

Bartell et al. [2] proposed a method that can automatically combine the estimates of multiple retrieval systems in order to search for the optimized parameter values. A standard vector-space retrieval system and a phrase identifier are combined in a linear model to obtain the overall estimate for document ranking problems. The objective of the optimization is to find parameter values such that the system can rank documents in a correct order.

Belkin el at. [3] combined a number of different query combinations of TREC topics into one single compound query for the information retrieval problem. Combination of the ranked lists of several retrieval results is applied. Results show that both combination of query representation and combination of ranked lists demonstrate overall better retrieval performance over both single query representation and single ranked list.

In addition to empirical findings, Lee [25] has also analyzed that improvements can be achieved with evidence combination since different runs might retrieve similar sets of relevant documents but retrieve different sets of nonrelevant documents.

Kivinen and Warmuth [16] has provided a theoretical work about the use of loss function on combining predictions from a set of on-line learning algorithms. The loss function is used for measuring the discrepancy between an expert's predictions and the actual observation. A weight is assigned for each expert, and at each iteration, the weights are updated with respect to the observed performance of that expert.

Voting is another common technique that aims to combine multiple evidence of different learning algorithms into a single prediction. Individual learning algorithms produce their own prediction for a particular learning task. The final prediction is made based on the majority vote for the prediction by all the algorithms. Specifically, a binary prediction is assumed to be a correct classification if the majority of classifiers predict for the binary prediction.

Voting can be divided into simple voting and weighted voting, while simple voting can be regarded as a special case of weighted voting. Contributions of different classifiers are considered to be equal under simple voting. For weighted voting, individual classifier is associated with a weight, which reflects the amount of its contribution of prediction towards the final prediction.

Littlestone and Warmuth [28] proposed the weighted voting algorithm for constructing a compound algorithm. The theoretical work assumes that the component classifiers make binary predictions. They have proved that there exists an upper bound for the number of mistakes the algorithm would make. The upper bound is proved to be a number that is smaller than the number of mistakes the best component classifier would make.

Chan and Stolfo [4] presented their evaluation of simple voting and meta-learning on partitioned data, through inductive learning. Particularly, they propose the use of an arbiter and a combiner strategy for the task of meta-learning. An arbiter is learned by all involved classifiers, and an arbitration rule is generated. The arbitration rule can affect the final prediction when breaking tie for the majority of vote, or the majority of classifiers do not agree. A combiner is used for coalescing the predictions from the involved classifiers by learning the relationship between the predictions they made and the correct prediction.

Ting and Witten [41] demonstrated the effectiveness of stacked generalization for combining different types of learning algorithms. Stacked generalization breaks the learning task into some low-level models and a high-level model. Low-level models are those general classification algorithms, which produce individual predictions on training data. These predictions, including the corresponding true classifications, are combined to form a new set of data, which is regarded as a new classification problem, for a learning algorithm, the high-level model.

Sohn [39] conducted studies on meta analysis of classification algorithms used for pattern recognition in the aspect of data mining. The proposed meta-model could predict expected classification performance of individual algorithms as a function of data patterns.

Nigam and Ghani [32] reported the co-training method of improving classification performance. Classifiers are separately trained by using a set of labeled documents. The classifiers are used for determining the confidence of a document belonging to a class. Such document is added to the set of labeled document of that class. The classifiers are applied to predict the class membership for the new documents. The predictions are finally combined together. However, its satisfactory experimental results rely on the independence of feature data sets split.

Kumar et al. [18] proposed a hierarchical multiclassifier system to perform hyperspectral data analysis. Ho [11] analyzed the complexity of classification problems using decision trees and nearest neighbour, and showed that dependences of classifiers' behaviour on data characteristics exist. Hull at al. [12] examined various combination strategies in the context of document filtering. Learning algorithms included Rocchio, nearest neighbor, linear discriminant analysis and neural net. Averaging strategies are studied and results show that overall filtering performance is improved.

Recently, there are meta-learning methods have been proposed specifically for text categorization domains.

Larkey and Croft [24] reported improved performance, by using new query formulation and weighting methods, in the context of text categorization by combining three classifiers linearly, namely k-NN, relevance feedback and Bayesian independence classifiers. Combination is based on either rank or scores of component classifiers. Improved performance is reported, however, the disadvantage of the proposed combination strategy rests on its manual calibration of the weight for each component classifier during combination.

Yang et al. [45] proposed the Best Overall Results Generator (BORG) system to reduce the variance of performance due to the lack of representative validation data sets in the Topic Detection and Tracking (TDT) domain. BORG combines classification methods linearly, using simple equal weight for each classifier. Classification methods employed in BORG are Rocchio, kNN and Language Modeling. The proposed combination strategy is a special case of equal weighting under linear combination strategy.

2.3 Our Meta-Learning Approaches

Most of the mentioned meta-learning approaches specifically for the task of text categorization mainly combine the classification decisions of different algorithms, without significant justification of the relative merit of using different component algorithms for different categories. Moreover, they are restricted with the involved component algorithms and model setting.

To tackle the inefficiency of the proposed methods, we introduce three different meta-learning approaches for text categorization, namely, the linear combination approach, the meta-learning using document feature characteristics, and the combination of these two approaches.

The Linear Combination approach can distill the characteristics of how we estimate the relative merit of each component algorithm for different categories. To determine the relative contribution of each component algorithms towards the final classification decisions, three different weight determining strategies are introduced. Different from the related work of meta-learning methods mentioned previously, our linear combination is not restricted to the involved number nor type of component classification algorithms.

By using document feature characteristics, our meta-learning approach, MUDOF, can automatically recommend an appropriate classification algorithm for each category, based on multivariate regression analysis by capturing category specific feature characteristics. Categorical feature characteristics are the descriptive summary about the specific nature and other specialties about a particular category. By learning the relationship between classification errors of different algorithms, and the categorical feature characteristics, it is hoped that the efficiency of classifying documents for a category by a certain classification algorithm can be estimated. According to the estimated performance, the approach can recommend the most suitable algorithm for each category. Using categorical document feature characteristics for recommending suitable classification algorithms has not been studied before. This approach is also not restricted by the number nor type of component classification algorithms.

The third approach is the combination of the other two proposed meta-learning methods. By incorporating MUDOF technique into the Linear Combination approach, MUDOF2 can derive the relative weight factors for each component classification algorithm with consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category.

Chapter 3

Document Pre-Processing

To conduct text categorization, we have to pre-process the textual documents into internal representation before constructing classifiers and perform the on-line classification. In this section, we discuss some background of pre-processing of text documents and the classification scheme learning strategy.

3.1 Document Representation

The classification system first extracts indexes or identifiers which can characterize the documents. Identifiers are basically words or phrases in the content and they can be used to represent the document. This indexing process is a pre-processing step before the system conducts document classification. In the past, indexing was mostly performed by subject experts, or some by well trained persons with experience in assigning content descriptions. However, manual indexing is very time consuming. Besides, indexing experts may introduce unwanted variability and uncertainties that may adversely affect the classification result and retrieval effectiveness. An alternative approach is based on *automatic indexing* [36, 34]. The main steps are described below:

1. Use a table, called the *stop-word list*, to eliminate common function words (eg. and, of, an, but, the, etc.) from the text documents.

2. Each of the remaining words is reduced to a word-stem form so that all words exhibiting the same stem are represented in the same way (eg. the words "analyze", "analyzes", and "analyzing" are all reduced to the stem *analy*).

3. Compute the term frequency f_{ij} for all stemmed words T_j in each document D_i.

As a result, each document is represented by a term vector of the form

$$D_i = (a_{i1}, a_{i2}, ..., a_{in})$$

where the coefficient a_{ik} represents the weight of the term T_k in document D_i.

These coefficients can be either binary or numeric. For the binary representation, a_{ik} is set to 1 when the term T_k is present in document D_i, and 0 when this term is absent. For numeric representation, the value of a_{ik} is determined from the effectiveness of this term to represent the document. Different kinds of numeric term weighting scheme have been proposed. One common method is the *term-frequency method* [35]. In this method, the value of a_{ik} is represented by the term frequency, f_{ik}, which is the number of occurrence of the term T_k in the document D_i. Thus we have

$$a_{ik} = f_{ik}$$

Another kind of representation is the *inverse document frequency method* [30, 35]. In this method, we need to obtain the inverse document frequency, I_k, of a term T_k which is defined as:

$$I_k = \log \frac{N}{d_k}$$

where N is the number of documents in a collection and d_k is the number of documents in a collection in which the term T_k occurs. The weight a_{ik} is determined by:

$$a_{ik} = f_{ik} I_k$$

Once a document is represented as a vector, the similarity between document D_i and D_j, $SIM(D_i, D_j)$, can be calculated in a number of ways. One popular method is the inner product as follows:

$$SIM(D_i, D_j) = \sum_{k=1}^{n} a_{ik} \cdot a_{jk}$$

The similarity coefficient is in principle unbounded, it is customary in most applications to use normalized similarity coefficients whose values vary between 0 and 1 when the vector elements are nonnegative. We adopt cosine coefficient as shown below: Cosine coefficient:

$$SIM(D_i, D_j) = \frac{2 \sum_{k=1}^{n} a_{ik} \cdot a_{jk}}{\sqrt{\sum_{k=1}^{n} a_{ik}^2 \sum_{k=1}^{n} a_{jk}^2}}$$

$$\begin{aligned}
D_1 &= \{a_{11}, a_{12}, ..., a_{1n} | L_1 : [c_{11}, c_{12}, ..., c_{1m}]\} \\
D_2 &= \{a_{21}, a_{22}, ..., a_{2n} | L_2 : [c_{21}, c_{22}, ..., c_{2m}]\} \\
&\vdots \\
D_t &= \{a_{t1}, a_{t2}, ..., a_{tn} | L_t : [c_{t1}, c_{t2}, ..., c_{tm}]\}
\end{aligned}$$

Figure 3.1: Internal representation of training documents collection

Some advantages of the vector representation are the model's simplicity, the ease with which it accommodates weighted terms, and its ability to rank the retrieved documents.

3.2 Classification Scheme Learning Strategy

To tackle the automatic classification problem, we make use of a machine learning technique which discovers classification knowledge or scheme, for each category, from a collection of training examples. Each example consists of a document and a set of manually assigned categories. Using the above representation, the training document collection can be represented as shown in Figure 3.1.

In Figure 3.1, a_{ij} denotes the weight of term T_j in document D_i and L_i denotes the set of labels assigned to document D_i. The value of c_{ij} indicates if category C_j is assigned to document D_i. The distribution of values of c are probably different from each other for different D, since each document can belong to any subset of the available per-defined categories. t is the total number of documents in the training document collection and n is the total number of indexed terms.

The classification scheme learning problem can be decomposed into subproblems related to individual categories. Since a fixed set of pre-defined categories is known in advance, we can learn a separate classification scheme for each category from the training document collection. After the classification schemes of all categories are discovered, they can be used together in the on-line classification module to decide a set of categories for a new document. Suppose the total number of pre-defined categories is m, and $m = \mid L_1 \mid = \mid L_2 \mid = \cdots = \mid L_t \mid$, where $\mid x \mid$ denotes the cardinality of x. For a document D_i and a particular category C_j, the value of c_{ij} is defined as:

$$c_{ij} = \begin{cases} 1 & \text{if } D_i \in C_j \\ 0 & \text{if } D_i \notin C_j \end{cases}$$

With the above representation, L_i is therefore a collection of values of 1 and 0, depending on whether the document D_i belongs to a category.

For example, if only documents D_1 and D_t in the training collection belong to the category C_j, then a particular representation of Figure 3.1 will be expressed as shown in Figure 3.2.

Using this training document collection, we can apply machine learning techniques to construct a classifier automatically for each category. Each term is regarded as a feature, and so each document is represented as a feature vector. After all categories have been learned, we have m classification schemes available for each classification algorithm. Each incoming new document is first converted into a system readable format and then matched

$$
\begin{aligned}
D_1 &= \left\{a_{11}, a_{12}, ..., a_{1n} | L_1 : [c_{11}, c_{12}, ..., c_{1j-1}, 1, c_{1j+1}, ..., c_{1m}]\right\} \\
D_2 &= \left\{a_{21}, a_{22}, ..., a_{2n} | L_2 : [c_{21}, c_{22}, ..., c_{2j-1}, 0, c_{2j+1}, ..., c_{2m}]\right\} \\
&\vdots \\
D_t &= \left\{a_{t1}, a_{t2}, ..., a_{tn} | L_t : [c_{t1}, c_{t2}, ..., c_{tj-1}, 1, c_{tj+1}, ..., c_{tm}]\right\}
\end{aligned}
$$

Figure 3.2: An example of internal representation of training documents collection

against each classification scheme. Each classification scheme outputs either a binary or weighted decision. Under our different proposed meta-learning approaches, these decisions are combined in various ways before making the final classification decision. Similar steps are repeated for each category and the system finally assigns a set of categories to each document.

Figure 3.3 shows a piece of sample document in Reuters-21578 document collection. The details of Reuters-21578 document collection will be given in Section 7.1. The documents are in SGML format. Each document starts with an "open tag" of the form <REUTERS ...> and end with an "close tag" of the form </REUTERS>. The list of topic categories for the document are enclosed by the tags <TOPICS> and </TOPICS>. If categories are present, each of them are delimited by the tags <D> and </D>. The main text of the document is enclosed by the tags <BODY> and </BODY>. The sample document shown in Figure 3.3 can be represented as:

$$D = \left\{a_{figur}, a_{regist}, ..., a_{linoil}, a_{show} | L : [c_{veg-oil} = 1, c_{linseed} = 1, ..., c_{wheat} = 1]\right\}$$

where $figur$, $regist$ and $linoil$ are the stemmed words selected from the document, while $veg - oil$, $linseed$ and $wheat$ are the subset of pre-defined category labels in the document collection.

```
<REUTERS TOPICS="YES" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET"
OLDID="5549" NEWID="6">
<DATE>26-FEB-1987 15:14:36.41</DATE>
<TOPICS><D>veg-oil</D><D>linseed</D><D>lin-oil</D><D>soy-oil</D>
<D>sun-oil</D><D>soybean</D><D>oilseed</D><D>corn</D>
<D>sunseed</D><D>grain</D><D>sorghum</D><D>wheat</D></TOPICS>
<PLACES><D>argentina</D></PLACES>
<PEOPLE></PEOPLE><ORGS></ORGS><EXCHANGES></EXCHANGES><COMPANIES></COMPANIES>
<TEXT>
<TITLE>ARGENTINE 1986/87 GRAIN/OILSEED REGISTRATIONS</TITLE>
<DATELINE>    BUENOS AIRES, Feb 26 - </DATELINE>
<BODY>Argentine grain board figures show crop registrations of grains,
oilseeds and their products to February 11, in thousands of tonnes,
showing those for future shipments month, 1986/87 total and 1985/86
total to February 12, 1986, in brackets:
    Bread wheat prev 1,655.8, Feb 872.0, March 164.6, total 2,692.4 (4,161.0).
    Maize Mar 48.0, total 48.0 (nil).
    Sorghum nil (nil)

    Oilseed export registrations were:
    Sunflowerseed total 15.0 (7.9)
    Soybean May 20.0, total 20.0 (nil)

    The board also detailed export registrations for subproducts, as follows,
    SUBPRODUCTS

    Wheat prev 39.9, Feb 48.7, March 13.2, Apr 10.0, total 111.8 (82.7).
    Linseed prev 34.8, Feb 32.9, Mar 6.8, Apr 6.3, total 80.8 (87.4).
    Soybean prev 100.9, Feb 45.1, MAr nil, Apr nil, May 20.0,
      total 166.1 (218.5).
    Sunflowerseed prev 48.6, Feb 61.5, Mar 25.1, Apr 14.5, total 149.8 (145.3)

    Vegetable oil registrations were :
    Sunoil prev 37.4, Feb 107.3, Mar 24.5, Apr 3.2, May nil,
      Jun 10.0, total 182.4 (117.6).
    Linoil prev 15.9, Feb 23.6, Mar 20.4, Apr 2.0, total 61.8, (76.1).
    Soybean oil prev 3.7, Feb 21.1, Mar nil, Apr 2.0, May 9.0,
      Jun 13.0, Jul 7.0, total 55.8 (33.7).
</BODY></TEXT>
</REUTERS>
```

Figure 3.3: A sample document in the Reuters-21578 document collection

Chapter 4

Linear Combination Approach

This chapter presents the overview and the algorithm framework of one of our proposed meta-learning approach, Linear Combination. Particularly, three different weighting strategies are proposed. We also compare the approach with some of the existing proposed methods of combining classifiers for text categorization. The comparison shows that the existing proposed methods are indeed special cases of one of the three weighting strategies under our Linear Combination approach.

4.1 Overview

There have been some research conducted to tackle the problem of text categorization by meta-learning techniques. As mentioned in Section 1.2, the goal of the meta-learning approach is to unify and combine the strength of the existing classification algorithms in order to achieve better overall classification performance.

A very simple method of combining the evidence of more than one classification algorithms is the majority voting as described in Section 2.2. However, giving the same amount of consideration for each involved component classifier when making the final classification decision may be undesirable, as what it does for majority voting. It is because we do not have a prior knowledge of the classification performance of each individual classifier towards a classification problem. For example, if a particular algorithm performs relatively worse, compared with other component classifiers, when predicting the membership of a certain category for documents, the general belief is that the influence of this algorithm to the final predictions of class membership should be less, or given little consideration, in order not to downgrade the overall classification performance.

A better approach to combine the evidence of different algorithms should be able to reflect and distill the characteristic of how we estimate the relative merit of each component algorithm for different categories under text categorization. Inspired by this idea, therefore, we propose the Linear Combination (LC) approach, that allows adjustment of influence, or the contributions, of the component classifiers towards the final prediction of class membership for incoming documents.

Figure 4.1 depicts the general framework of our linear combination approach for a particular category. Component classifiers are individually constructed by some learning algorithms for a particular category. The learning algorithms are not bound to be same type nor same nature. Confidence scores of class membership for each incoming document are calculated by each component classifier and generate its own scores distribution. For each document, the component scores are combined by a linear combination scheme and it yields the final scores distribution. Based on this new scores distribution, documents associated with confidence scores higher than a threshold value, t, are classified as a member of the category. One important issue is the weighting strategy used in the linear combination scheme. We investigate three different weighting strategies in our research.

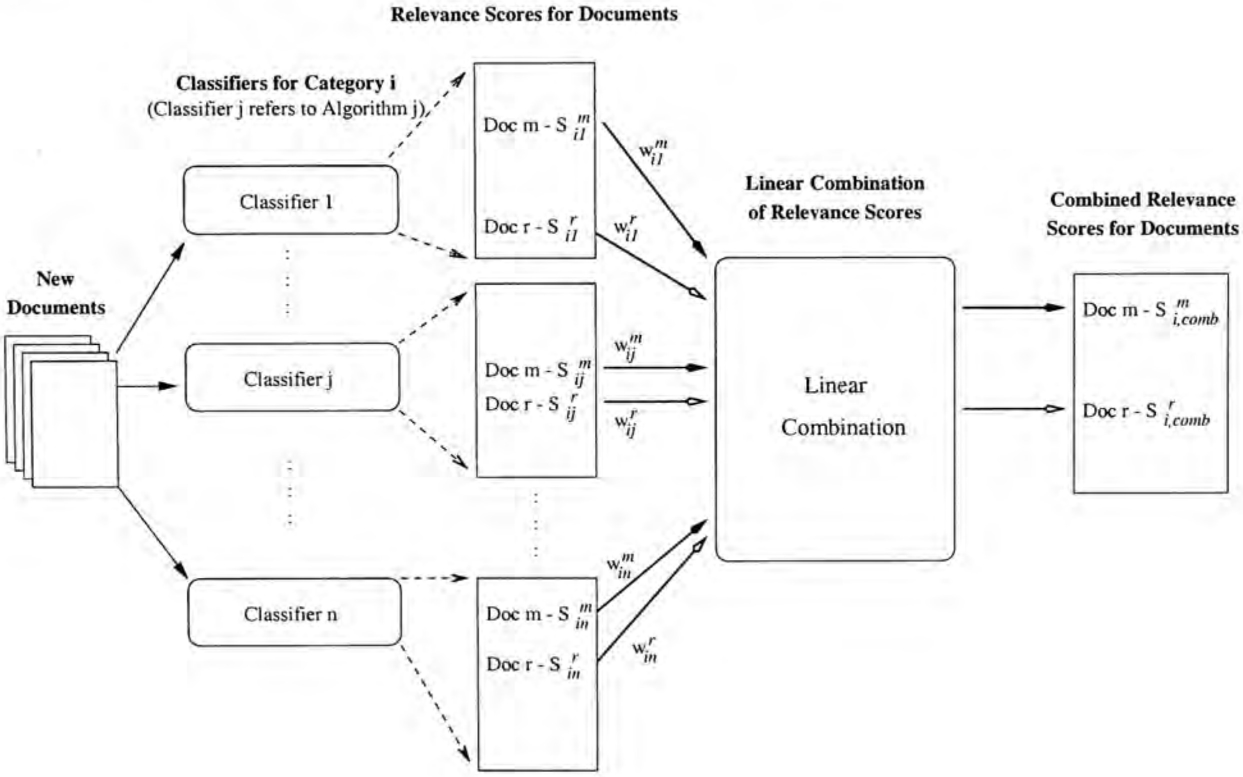

Figure 4.1: The framework of applying linear combination technique for a particular category

The following sections present the detailed algorithm of our proposed linear combination approach. Particularly, three different weighting strategies determining the contributions of the component classifiers are described.

4.2 Linear Combination Approach - The Algorithm

We first present a general linear combination approach. Consider a particular category i. The contribution of each individual component algorithm j to the final meta-model classification decision is represented by a weight factor w_{ij}. Suppose there is a document m which is to be categorized. Instead of using the relevance score calculated from a single classification scheme of a particular category, the linear combination approach calculates a combined score which is the weighted sum of contributions of all component algorithms in a linear fashion. Suppose there are n component algorithms. The combined score for m is computed by Equation 4.1.

$$S^m_{i,comb} = \sum_{j=1}^{n} \omega^m_{ij} * S^m_{ij} \tag{4.1}$$

where $S^m_{i,comb}$ is the final combined relevance score for m in the category i. S^m_{ij} is the score calculated between m and the classifier learned by algorithm j for category i. The value of ω^m_{ij} is the weight factor, or the contribution, of the classifier to the score S^m_{ij}, and $\sum_{j=1}^{n} \omega^m_{ij}$ is equal to 1.

If the final combined score for m is larger than the threshold value set for a category, that category is assigned to m. Figure 4.1 depicts the overall framework of the linear combination approach To reflect the significance of contribution by different classifiers for a category, various strategies can be employed to determine the weight (ω^m_{ij} in Equation 4.1). We have proposed three weighting strategies under this linear combination approach, to study the categorization performance differences.

4.2.1 Equal Weighting Strategy

The first strategy, called LC1, is an equal weighting scheme. Under this scheme, the weight of all classifiers are the same, as indicated in Equation 4.2. As a result, the contribution of each classification algorithm to the final combined score for m is equal.

$$\omega_{i1}^{m} = \omega_{i2}^{m} = \cdots = \omega_{ij}^{m} = \cdots = \omega_{in}^{m} = \frac{1}{n} \qquad \text{for all } i \tag{4.2}$$

4.2.2 Weighting Strategy Based On Utility Measure

The second strategy, called LC2, determines the weights based on utility measure from training. Under this strategy, the relative contribution, w_{ij}^{m}, of a classification scheme, which is constructed by algorithm j for category i, to the final combined score for document m, depends on the performance, u_{ij}, of the learned classifier in the training phase. The relationship between the contribution of the classifier and its categorization performance, is represented as a function indicated in Equation 4.3.

$$\omega_{ij}^{m} = f(u_{ij}) \tag{4.3}$$

where function f is expressed in terms of u_{ij}, which is the utility score obtained by the classifier.

The function f is a transformation function from certain utility scores to corresponding contribution weights. The transformation is restricted by the condition that $\sum_{j=1}^{n} \omega_{ij}^{m}$ equals to 1. Conceptually, a well-performed classifier constructed by an algorithm should be given a heavier weight than the others during the score combination. In our investigation, we adopt the function f

as shown in Equation 4.4.

$$\omega_{ij}^{m} = f(u_{ij}) = \frac{u_{ij}}{\sum_{k=1}^{n} u_{ik}} \qquad \text{for } 1 \leq j \leq n \tag{4.4}$$

We make use of a set of documents, called tuning set, obtained from a subset of the training set to calculate u_{ij}. Specifically, u_{ij} is the classification performance of the tuning set using the classification scheme constructed by algorithm j for category i.

4.2.3 Weighting Strategy Based On Document Rank

Our third strategy, called LC3, determines the contribution weights of the involved component algorithms, based on the rank of scores, S_{ij}^{m} for document m. The scores for document m are first ranked across the component algorithms. By mapping from the rank R to a set of pre-determined weight factors using the function g, a particular weight, say P_d, is assigned to the corresponding algorithm as its contribution in the final combined score for m. The idea of this strategy is illustrated in Equation 4.5.

$$\omega_{ij}^{m} = P_d = g(R_{ij}^{m}) \qquad \text{for } P_d \in \{P_1, P_2, \ldots, P_n\} \text{ and } 1 \leq j \leq n \tag{4.5}$$

where P_d is one of the n pre-determined weights, and R_{ij}^{m} is the rank of score of m by algorithm j under category i. g is a mapping function from the rank R_{ij}^{m} to the assignment of the weight P_d for the document m. For example, if the score rank R_{ij}^{m} is the second highest among other classifiers and so its value is equal to 2, then the weight assigned to algorithm j for the combination with other classifiers for document m in category i is P_2.

4.3 Comparisons of Linear Combination Approach and Existing Meta-Learning Methods

Our proposed linear combination approach is a more general and formalized version of some existing meta-learning methods for text categorization, namely, the simple majority voting, BORG proposed by Yang et al. [45] and the restricted combination method proposed by Larkey and Croft [24].

4.3.1 LC versus Simple Majority Voting

As described in Section 2.2, the final classification prediction for simple majority voting is made based on the plurality of vote for the prediction by all the involved component algorithms. Indeed, this combination approach is a special case of our generalized linear combination approach with equal weighting strategy.

For example, by simple majority vote, if there are three, out of five, classifiers voting for the class membership of a category i for a document m, then that document is believed to belong to that category since more than half ($3/5 = 0.6 > 0.5$) of the component algorithms agree with the predictions. In this case, 0.5 is a fixed threshold for the category under consideration, and 0.6 is treated as the final confidence score for the document. Since the final score is larger than the threshold, the document is assigned with the category label.

We attempt to show that simple majority voting is a special case of equal weighting strategy. Consider that each individual algorithm's vote can be represented as a real discrete value. The vote carries the score value of 1 if the algorithm vote for a prediction of class membership, and 0 if otherwise. Also, we assign an equal and real-valued weight for each algorithm's vote, and the summation of these weight should be equal to 1. Consequently, $\omega_{ij}^{m} = 1/n$. The threshold t is set to 0.5. Returning to the previous example, if we use the notations introduced in Section 4.2:

$$n = 5 \qquad and \qquad S_{i1}^{m} = S_{i2}^{m} = S_{i3}^{m} = 1 \qquad and \qquad S_{i4}^{m} = S_{i5}^{m} = 0$$

Also, we assign an equal and real-valued weight for each algorithm's vote, and the summation of these weight should be equal to 1 as shown below:

$$\omega_{i1}^{m} = \omega_{i2}^{m} = \omega_{i3}^{m} = \omega_{i4}^{m} = \omega_{i5}^{m} = \frac{1}{5} = 0.2$$

Then, the value of the final confidence score under our linear combination approach can be calculated by Equation 4.6

$$S_{i,comb}^{m} = 0.2 * 1 + 0.2 * 1 + 0.2 * 1 + 0.2 * 0 + 0.2 * 0 = 0.6 \tag{4.6}$$

Since $S_{i,comb}^{m} > t$, category i is assigned to document m. The trivial and simple proof shows that majority voting can indeed be considered to a special case for our proposed generalized Linear Combination method with equal weighting strategy.

4.3.2 LC versus BORG

We have briefly described the work of BORG in Section 2.2. In order to improve the consistency of statistical classifiers when lacking representative validation sets under the TDT domain, BORG combines each classifier by simple summation. Specifically, the scores of documents in each experimental run of each classifier are first normalized by the mean and standard deviation of the scores. The normalized scores of each document for each classifier are then summed together. The final combined scores for all documents are finally re-normalized in the same way to produce the final scores. Such simple summation can therefore be considered to be the case of equal weighting strategy under our Linear Combination approach, since the individual component scores are summed together without considering the relative contributions. Therefore, each algorithm is regarded to contribute equally ($\omega_{ij}^{m} = \frac{1}{n}$ for all categories and classification algorithms) towards the final predictions.

4.3.3 LC versus Restricted Linear Combination Method

The linear combination method proposed by Larkey and Croft [24], as introduced in Section 2.2 is a restricted version of our generalized Linear Combination approach. Their work mainly studied the combination of KNN, Bayesian Independence Classifiers and Relevance Feedback ($n = 3$). Studies of using additional component algorithms during combination has not been given. The associated weight for each component algorithm is *manually* tuned, which may not be able to uncover the robustness of the linear combination approach.

Instead, our generalized linear combination approach can flexibly adapt to different number of component algorithms. In addition to manual assignment of associated weight for each algorithm, our method can also automatically assign weight for each component algorithm according to the preliminary classification performance, under the Weighting Strategy Based On Utility Measure (LC2) and the Weighting Strategy Based On Document Rank (LC3).

Chapter 5

The New Meta-Learning Model - MUDOF

Most existing meta-learning approaches for text categorization are based on linear combination of several basic algorithms. The linear combination approach makes use of limited knowledge in the training document set. To address this limitation, we propose a meta-model approach, called Meta-learning Using Document Feature characteristics (MUDOF), which employs a meta-learning phase using document feature characteristics. Document feature characteristics, derived from the training document set, capture some inherent category-specific properties of a particular category. This approach aims at recommending a suitable algorithm automatically for each category. Hence, different algorithms may be employed for constructing classifiers for different categories. Specifically, the relationship between the document feature characteristics and the predicted classification error of a classification algorithm is learned by using the technique of multivariate regression analysis. Based on the relationship, it can make automatic recommendation of algorithms for different categories. As a result, our MUDOF approach combines the evidence of predicted classification errors of different algorithms by regression analysis on document feature characteristics.

5.1 Overview

While the improvements reported by most of the previous studies of different approaches on text categorization were evaluated based on several single overall performance scores calculated by different utility measures, however, performance comparisons, on category-by-category basis, between different algorithms are seldom investigated. We observe that, though a particular algorithm may obtain a better overall performance in different single performance scores, it is not guaranteed that its performance is the best for particular categories, when compared with other algorithms. In fact, given a certain category, the classification performance varies with the choice of algorithms. This can be attributed to the fact that algorithms perform differently for a certain category, which exhibits specific nature or different characteristics from other categories. If an algorithm, of less capable in classification performance, for a particular category, can be replaced by another efficient algorithm, an overall better classification performance can be further increased.

Motivated by such observations, we propose MUDOF, a novel approach of the meta-learning framework for text categorization, based on multivariate regression analysis, by capturing category specific feature characteristics.

Different from existing categorization methods, instead of applying one single method for all categories during classification, this new meta-learning approach can automatically recommend a suitable algorithm during training, from an algorithm pool, for each category based on the category specific statistical characteristics and multivariate regression analysis. We employ a meta-learning approach by learning the relationship between the feature characteristics and the classification errors by conducting multivariate regression analysis for each algorithm on each category. The learned relationship is expressed by sets of parameter estimates, based on which, suitable classification algorithms are recommended for the categories. Document feature characteristics, derived from the training set of a particular category, can capture some inherent properties of that category. The problem of predicting the expected classification error of an algorithm for a category, therefore, can be interpreted as a function of these feature characteristics. In summary, our proposed MUDOF approach consists of three key components, namely categorical document feature characteristics, classification errors and a multivariate regression model. Figure 5.1 shows the overview of our proposed approach. The following sections will present the details of these three key components.

5.2 Document Feature Characteristics

Document feature characteristics, derived from the training set of a particular category, can capture some inherent properties of that category. They are the statistics that can be regarded as the descriptive summary for documents

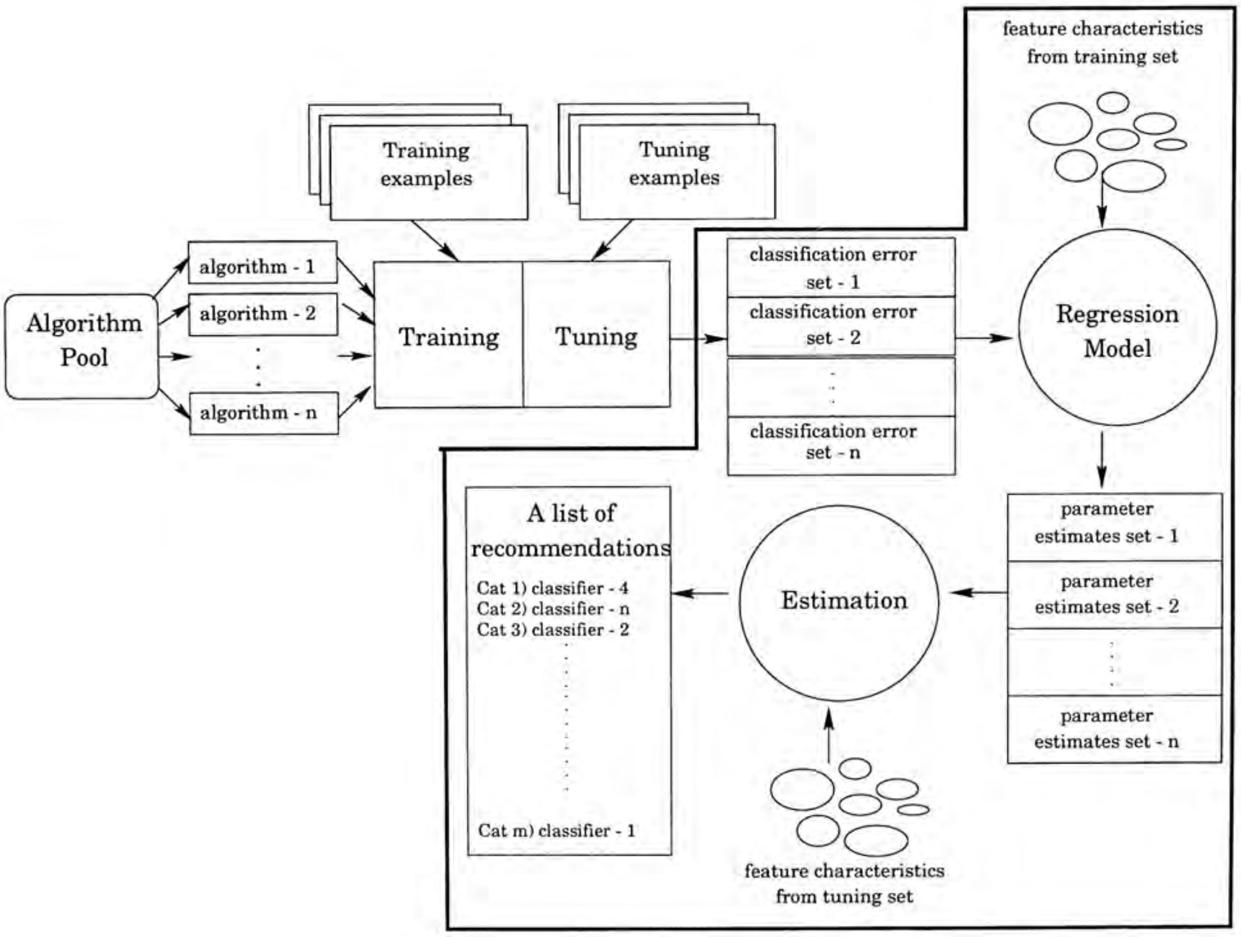

Figure 5.1: An overview of the meta-model approach for text categorization

belonging to a certain category.

Study on term feature selection has been done [49] to reduce the dimensionality of feature space aggressively. The study concentrates on selecting useful and informative term features by various means. However, selecting useful feature terms is limited to capture the contextual information only, while other aggregate or higher level characteristics a category exhibits may not be revealed. Instead of merely capturing the contextual information of a category, our MUDOF approach also consider the general nature or specific characteristics for the documents belonging to a category. These characteristics, collected on category basis, are not necessarily directly related to the context of a document of a category. We believe that, capturing the specific characteristics in addition to the contextual information of each category, can help revealing the relationship between the document feature characteristics and the classification performance of different algorithms applied. Some examples of document feature characteristics are given below:

1. ***PosTr***: The number of positive training examples of a category.
2. ***AvgDocLen***: The average document length of a category.

A complete list of document feature characteristics used in our investigation is given in Section 7.4.

5.3 Classification Errors

Classification errors directly reflect the classification performance of classifiers against a category. The general belief is that an algorithm is considered to perform better if it has a smaller classification error when compared with other classifiers solving the same classification problem. Given the same set of documents, different algorithms can be used for training the corresponding classifiers for a particular category. As the constructed classifiers are different, therefore, different classifiers generally demonstrate different ability of predicting the class membership for a category.

Existing classification algorithms attribute significant classification errors to the imperfect quality of a classifier. In addition to the quality of classifiers, our MUDOF approach relates the classification errors of an algorithm to the appropriateness of classifying documents for a category that exhibit specific characteristics. In general, if a classification algorithm can handle the classification task better for certain types, or categories, of documents, the predicted classification error for the algorithm should be the smallest among the others, and so the algorithm should be recommended to handle the classification of future documents for the categories. As the classification errors are directly related to which algorithm is applied, they are fitted into the regression model as dependent variables under our MUDOF approach.

5.4 Linear Regression Model

Regression analysis is a statistical technique for modeling and investigating the inherent relationship between two or more variables. For example, the prediction accuracy of a stock price may be considered to be related to the time frame of the history data that is available for making the prediction. Regression analysis can be applied to build a mathematical model to predict the accuracy at a given time frame level in the form as shown in Equation 5.1.

$$Y = \beta_0 + \beta_1 * x + \epsilon \tag{5.1}$$

The model shown in Equation 5.1 is called the simple linear regression model, in which there is only one response variable, or the dependent variable, Y, and only one regressor, or the independent variable, x. βs are the regression coefficients, or the parameter estimates, that depicts the relationship between the independent variable and dependent variables. ϵ is the random error term. The term *linear* is used because Equation 5.1 is a linear function of the unknown parameters.

To construct the regression model, we have to collect a number of observations (x_1,y_1), (x_2,y_2), ..., (x_n,y_n) and express them as Equation 5.2. The value of β_0 and β_1 can be estimated by using the method of least squares estimation. The final estimated regression model is expressed as Equation 5.3. We can use the estimated regression model to make future prediction of the value of $\widehat{y}$ by fitting a given value of x into the model.

$$y_i = \beta_0 + \beta_1 * x_i + \epsilon_i \qquad i = 1, 2, ..., n \tag{5.2}$$

$$\widehat{y} = \widehat{\beta_0} + \widehat{\beta_1} * x \tag{5.3}$$

Multiple linear regression model is a more generalized regression model in which more than one independent variables are included in the regression model as shown in Equation 5.4.

$$Y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \ldots + \beta_k * x_k + \epsilon \tag{5.4}$$

Similar to simple linear regression, least squares estimation is also used to construct the regression model, except that multiple regression model now involves more than one independent variables during the estimation. The final estimated model is expressed in the form as shown in equation 5.5

$$\widehat{y} = \widehat{\beta_0} + \widehat{\beta_1} * x_1 + \widehat{\beta_2} * x_2 + \ldots + \widehat{\beta_1} * x_k \tag{5.5}$$

Multiple linear regression model, or multivariate regression analysis, is usually used for making predictions for problems which involve more than one factors that contribute to the final predictions. We choose to employ multivariate regression analysis for our MUDOF approach since documents belonging to a category usually exhibits more than one particular nature or specialties. Therefore, considering more than one document feature characteristics for each category when making algorithm recommendations can capture the more complete and useful information of each category.

5.5 The MUDOF Algorithm

In MUDOF, we make use of categorical document feature characteristics and classification errors. In particular, we wish to predict the classification error for a category based on the feature characteristics. This is achieved by a meta-learning approach based on regression model, in which, the document feature characteristics are the independent variables, while the classification error of an algorithm is the dependent variable. We further divide the training collection into two sets, namely the training set and the tuning set. Two sets of feature characteristics are collected separately from these two data sets. Statistics from the training set are used for parameter estimations. Together with the estimated parameters, the feature characteristics from the tuning set are used for predicting the classification error of an algorithm for a category. The algorithm with the minimum estimated classification error for a category will be recommended for that category during the on-line classification, or validation phase.

Consider the ith category. Suppose we have several component classification algorithms. Let e_{ij} be the classification error of the training set on the jth algorithm. Classification errors will first undergo a logistic transformation to yield the response variable, or the dependent variable, for the meta-model. Precisely, the transformation is given in Equation 5.6.

$$y_{ij} = \ln \frac{e_{ij}}{1 - e_{ij}} \tag{5.6}$$

where y_{ij} is the response variable. This transformation ensures that the response variable is in the range of 0 and 1. The response variable, y_{ij} is related to the feature characteristics by the regression model, as shown in Equation 5.7.

$$y_{ij} = \beta_j^0 + \sum_{k=1}^{p} \beta_j^k * F_i^k + \epsilon_{ij}, \tag{5.7}$$

where F_i^k is the kth feature characteristic in the ith category. The number of document feature characteristics used in the meta-model is p. β_j^k is the parameter estimate for the kth feature, by using the algorithm j. ϵ_{ij} is assumed to follow a Gaussian distribution $N(0,var(\epsilon_{ij}))$. Based on the regression model above, the outline of meta-model for text categorization is given in Figure 5.2.

Step 1 to 9, in Figure 5.2, aim to estimate a set of betas ($\widehat{\beta_j^k}$), the parameter estimates of the feature characteristics in the regression model, for each individual algorithm. In Step 2, an algorithm, with optimized parameter settings, is picked from the algorithm pool. By repeating Steps 3 to 7, the algorithm is applied on training and tuning examples to yield classification errors of the classifier for all categories. Documents in tuning set, as shown in Step 5, are used for obtaining the classification performance, and so the classification error, of a trained classifier for each category. A set of betas, belonging to the algorithm being considered, can be obtained by fitting all classification errors of the categories, and their corresponding feature characteristics in the training set, into the regression model. After Step 9, there will be n sets of estimated parameters, the betas, each of which corresponds to the relationship between the classification performance of a component classification algorithm and the document feature characteristics. These estimated parameters are then used for the subsequent steps.

The predictions on the classification errors of the involved algorithms are made from Steps 10 to 16. In Step 12, one algorithm with the same optimized parameter settings as in Step 2, is picked from the algorithm pool. The corresponding set of betas of the selected algorithm, together with the feature characteristics of a category in the tuning set, will be fitted into the regression model, in Step 13, to give the estimated classification errors of the algorithm on the category. Decisions, about which algorithm will be applied on the category, are based on the predicted minimum classification errors in Step 14. After Step 16, classification algorithms are recommended for categories, and the recommended algorithm will be applied to each category during the on-line classification, or validation.

The robustness of the meta-model approach rests on its fully automatic estimations. The whole process, from parameter estimation to recommending algorithms for categories, is fully automatic. The operation of our meta-model approach is carried out as usual, except that different algorithms will be applied to the categories, instead of applying a single algorithm on all categories as what is commonly done in other approaches.

```
Input:  The training set $TR$ and tuning set $TU$
        An algorithm pool $A$ and categories set $C$

Repeat
    Pick one algorithm $ALG_j$ from $A$.
    For each category $C_i$ in $C$
        Apply $ALG_j$ on $TR$ for $C_i$ to yield a classifier $CF_{ij}$.
        Apply $CF_{ij}$ on $TU$ for $C_i$ to yield classification error $e_{ij}$.
        Take logistic transformation on $e_{ij}$ to yield $y_{ij}$ for later parameter estimation.
    End For
    Estimate $\widehat{\beta}_j^k$ ($k$=0,1,2,...,p) for $ALG_j$ by fitting $y_{ij}$ and $F_i^k$ (in $TR$)
      into the regression model.
Until no more algorithms in $A$.
For each category $C_i$ in $C$
    Repeat
        Pick one algorithm $ALG_j$ from $A$.
        Estimate the classification error $\widehat{e}_{ij}$ by fitting $\widehat{\beta}_j^k$ and corresponding $F_i^k$ (in $TU$)
          into the regression model.
        If $\widehat{e}_{ij}$ is minimum, recommend $ALG_j$ for $C_i$ as the output.
    Until no more algorithms in $A$.
End For
```

Figure 5.2: The MUDOF algorithm

Chapter 6

Incorporating MUDOF into Linear Combination approach

After studying our two proposed meta-learning methods, namely the Linear Combination approach and MUDOF in Chapter 4 and Chapter 5 respectively, we present an approach for further improving the Linear Combination approach. We first discuss the motivation behind our proposed method and then introduce the overall framework of the approach in details.

6.1 Background

Combining classification evidence by different linear combination approaches as mentioned in Section 2.2 demonstrate that classification performance can be improved by considering the classification decisions of different component algorithms before making the final prediction. Our first proposed approach is Linear Combination approach which is a more generalized version of other existing linear combination strategies. Instead of regarding each component classifier's classification decisions as equal, our proposed approach is able to reflect and distill the characteristic of how we estimate the relative merit of each component algorithm for different categories under text categorization. Specifically, the contribution of each individual component classification algorithm is reflected by a weight factor, which is determined based on a classifier's preliminary classification performance available. If the preliminary classification performance of an algorithm is better than the others, then we have a ground to believe that the classification decisions made by such algorithm during on-line classification should be given more consideration towards the final decision to be made.

In addition to combining multiple evidence of different classification algorithms under linear combination approach, considering the specific nature and knowledge of characteristics about a category is also beneficial to the improvement of classification performance. As shown in Figure 4.1, the final output for linear combination approach is a combined distribution of confidence scores related to the relevance of documents to a class membership, without considering the complete knowledge associated with a category. We believe that, by capturing useful information of categorical document feature characteristics and integrating them into the Linear Combination approach, better classification performance should be observed.

Our proposed second approach is MUDOF which employs a meta-learning phase using document feature characteristics which are collected on category basis and therefore can capture some inherent properties of each category. By learning the relationship between categorical document feature characteristics and classification errors by multivariate regression analysis, the approach can recommend the most suitable algorithm for each category. The relationship learned between document feature characteristics and classification errors can be used for calculating the predicted classification errors of different classification algorithms. The one with the smallest predicted classification error is recommended for a category. As a result, the MUDOF approach is able to learn a more comprehensive knowledge of a category before making the algorithm recommendation.

Since our MUDOF approach can recommend classification algorithms for each category based on the learned relationship between document feature characteristics and classification errors, the technique indeed can be integrated into our proposed Linear Combination approach, so that the combination of the two approaches results in a new method called MUDOF2. Specifically, in MUDOF2, we would like to derive the relative weight factors for each component classification algorithm with proper consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category.

6.2 Overview of MUDOF2

MUDOF2 integrates the technique of MUDOF and the Linear Combination approach. Instead of recommending one single classification algorithm for each category, MUDOF2 now considers and combines the classification decisions across all component classification algorithms. There are two major steps involved in MUDOF2. The first step is to employ a modified MUDOF approach to generate a matrix of estimated classification performances. The second step is to employ our proposed Linear Combination approach to combine the individual confidence scores of different classification algorithms to yield a final confidence score belonging to a class membership for a document. The original MUDOF approach is modified so that it can output the matrix of estimated classification performances. Figure 6.1 depicts the framework of MUDOF2. Similar to the framework introduced in Section 5.1, we employ a meta-learning approach by learning the relationship between the feature

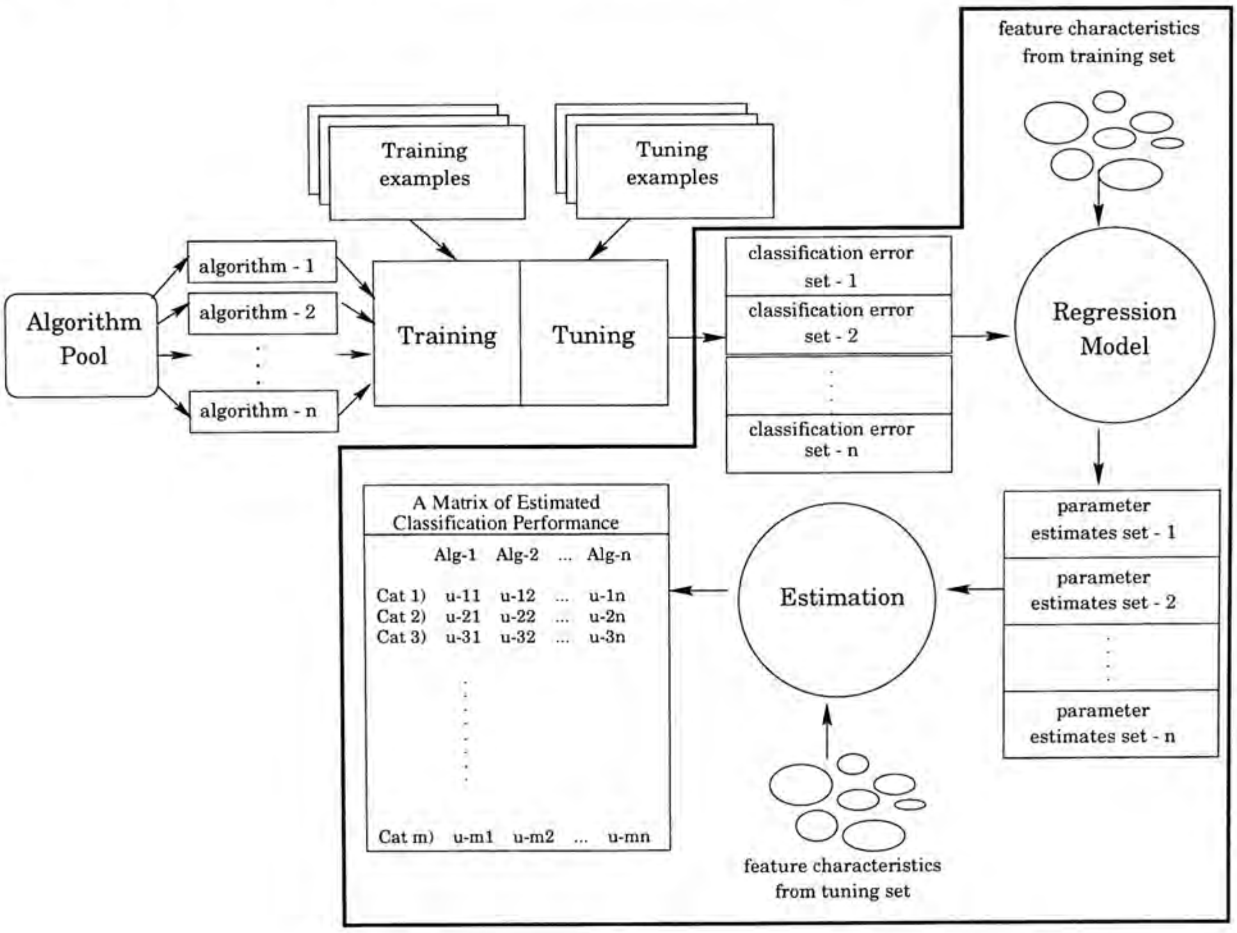

Figure 6.1: An overview of MUDOF2

characteristics and the classification errors by conducting multivariate regression analysis for each algorithm on each category. However, instead of recommending one single algorithm from the algorithm pool, MUDOF2 predicts the classification errors, and so the classification performances, for each classification approaches against each category and form a matrix of estimated classification performance. The scores matrix consists of performance scores which are estimated by considering the categorical document feature characteristics.

Recall that in our Linear Combination approach, we have proposed three weighting strategies as discussed in Section 4.2. Each of the proposed strategies has a different perspective on the contributions of the involved component classifier towards the final classification decisions to be made. To incorporate the consideration of document feature characteristics in order to learn a more comprehensive knowledge about the nature of a category, we enhance the Linear Combination approach by making use of the estimated performance scores matrix to determine the weight factors, or the relative contributions, for each component classification approach. Particularly, the Weighting Strategy Based On Utility Measure as proposed in Section 4.2.2 is employed to determine the weight factors. The weight factors calculated with the estimated scores matrix are used for combining the classification decisions. The weight factors correspond to those ω as depicted in Figure 4.1.

It should be noted that, since the Equal Weighting Strategy as introduced in Section 4.2.1 regards the contribution of each component classification algorithm as equal, performance obtained after integrating the additional step of generating estimated scores matrix into the original Linear Combination approach should be identical to the results that would have produced by using the original Linear Combination approach. As a result, the Equal Weighting Strategy is not employed.

Besides, the Weighting Strategy Based On Document Rank as proposed in Section 4.2.3 is concerned with individual ranks for each document under different classification algorithms. Since MUDOF concerns with document feature characteristics and classification errors on a category basis instead of individual document ranks, the Linear Combination approach with the Weighting Strategy Based On Document Rank is not useful after being integrated with the additional step of predicting classification performance based on categorical document feature characteristics. As a result, the Weighting Strategy Based On Document Rank is not employed.

In summary, MUDOF2 integrates our MUDOF and our Linear Combination approach particularly with the Weighting Strategy Based On Utility Measure. We would like to study the effect on the classification performances by combining the classification decisions of different classification algorithms and by capturing the inherent properties of each category, with the available categorical document feature characteristics.

6.3 Major Components of the MUDOF2

The key components of the modified approach include categorical document feature characteristics, classification errors, a multivariate regression model and relative weight factors.

Under the approach, categorical document feature characteristics are collected in a similar way as our proposed MUDOF approach as described in Chapter 5. The document feature characteristics are collected on a category basis and therefore they are regarded as descriptive summary for documents belonging to a certain category.

Classification errors of different classification algorithms are also collected on category basis. Different classification algorithms can be employed to construct the corresponding classifiers for each category. Due to the different nature of different classifiers, they usually demonstrate different classification performance for the same category. Instead of recommending one single algorithm of minimum predicted classification error for a category, this modified approach considers the predicted errors of all involved component classification algorithms. According to the predicted classification errors, an estimated classification performance scores matrix is constructed. The matrix is used for calculating the relative weight factors for each component classification algorithm for linear combination.

The multivariate regression model used is the same as that proposed for MUDOF in Section 5.5. It serves the function of learning the relationship between the categorical document feature characteristics and the classification errors of different classification algorithms. The set of parameter estimates calculated are regarded as the learned relationship. Based on this relationship, the classification errors are predicted and a matrix of estimated classification performance of different classification algorithms is generated. The estimated classification performance is therefore derived by capturing the inherent properties of each category based on the learned relationship.

Based on the generated matrix of estimated classification performance and the Weighting Strategy Based On Utility Measure as proposed in Section 4.2.2, the relative weight factors for each component classification algorithm are calculated on category basis. The relative weight factors represent the relative contribution of the classification decisions of individual classifiers towards the final classification decision. The weight factors are used for combining the confidence scores of documents across the involved component classification algorithms to yield a final combined score, which is then used for determining if a category label should be assigned to the document with respect to a threshold.

It should be noted that the weight factors derived under this modified approach are of different nature from that determined in Section 4.2.2. In Section 4.2.2, the weight factors are determined based on the preliminary classification performance of the component classification algorithms only. In addition to the classification performance, under MUDOF2, the weight factors are determined with consideration of documents' specific nature and characteristics of each particular category. As a result, the weight factors are incorporated with more comprehensive knowledge of the properties of each category.

6.4 The MUDOF2 Algorithm

MUDOF2 combines the classification decisions of all involved component classification algorithms based on the estimated classification performance, instead of recommending one single classification algorithm for each category as proposed in our MUDOF approach. To achieve this, we mainly follow the modified framework of MUDOF and integrate with the Linear Combination approach. MUDOF2 involves two major steps. The first step is to generate a matrix of estimated classification performance of different algorithms against each category. The second step is to determine the relative weight factors associated with each component classification algorithm for the linear combination of classification decisions.

To take the advantage of considering categorical document feature characteristics when determining the relative weight factors for each component classification algorithm, we make use of categorical document feature characteristics and classification errors. Similar to MUDOF approach, we wish to predict the classification error, and so the classification performance for a category based on the document feature characteristics. This is achieved by a meta-learning approach based on regression model, in which, the document feature characteristics are the independent variables, while the classification error of an algorithm is the dependent variable. We divide the training collection into the training set and the tuning set. Two sets of feature characteristics are collected separately from these two data sets. Statistics from the training set are used for parameter estimations. Together with the estimated parameters, the statistics from the tuning set are used for predicting the classification error of an algorithm on a category. Instead of recommending the algorithm with the minimum estimated classification error for a category during the on-line classification, the approach now considers the estimated classification performance of all classification algorithms and generate a matrix consisting of all the estimated classification performance.

Suppose there are m categories and n algorithms, the estimated performance matrix P is expressed as a $m * n$ matrix as shown in Equation 6.1.

$$\mathbf{P} = \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1j} & \dots & u_{1n} \\ u_{21} & u_{22} & \dots & u_{2j} & \dots & u_{2n} \\ \vdots & \vdots & & \vdots & & \vdots \\ u_{m1} & u_{m2} & \dots & u_{mj} & \dots & u_{mn} \end{pmatrix} \tag{6.1}$$

where u_{ij} refers to the estimated classification performance, a particular element inside P, for classification algorithm j against category i.

The matrix is then used for determining the relative weight factors of the component classification algorithms for the linear combination of classification decisions. Particularly, we employ the Weighting Strategy Based On Utility Measure as proposed in Section 4.2.2 to calculate the weight factors. During the on-line classification, confidence scores of a class membership for a document are combined across all the involved component classification algorithms with the use of pre-determined relative weight factors. The final combined confidence scores are used for determining the class membership for the document with respect to a determined threshold value.

Consider the ith category. Suppose we have several component classification algorithms. The classification error, e_{ij}, of the training data set on the jth algorithm is first obtained by Equation 6.2.

$$e_{ij} = 1 - u_{ij} \tag{6.2}$$

where u_{ij} is the estimated performance of the algorithm j against category i in the performance matrix shown in Equation 6.1. The classification error will then undergo a logistic transformation to yield the response variable, or the dependent variable, for the meta-model as shown in Equation 6.3.

$$y_{ij} = \ln \frac{e_{ij}}{1 - e_{ij}} \tag{6.3}$$

where y_{ij} is the response variable. The response variable, y_{ij} is related to the feature characteristics by the same regression model as MUDOF, as shown in Equation 6.4.

$$y_{ij} = \beta_j^0 + \sum_{k=1}^{p} \beta_j^k * F_i^k + \epsilon_{ij}, \tag{6.4}$$

where F_i^k is the kth feature characteristic in the ith category. The number of document feature characteristics used in the meta-model is p. β_j^k is the parameter estimate for the kth feature, by using the algorithm j. ϵ_{ij} is assumed to follow a Gaussian distribution $N(0,var(\epsilon_{ij}))$. Based on the regression model above, the outline of meta-model for text categorization is given in Table 6.1.

```
Input:  The training set $TR$ and tuning set $TU$
        An algorithm pool $A$ and categories set $C$

Repeat
    Pick one algorithm $ALG_j$ from $A$
    For each category $C_i$ in $C$
        Apply $ALG_j$ on $TR$ for $C_i$ to yield a classifier $CF_{ij}$
        Apply $CF_{ij}$ on $TU$ for $C_i$ to yield classification error $e_{ij}$
        Take logistic transformation on $e_{ij}$ to yield $y_{ij}$ for later parameter estimation
    End For
    Estimate $\widehat{\beta}_j^k$ ($k$=0,1,2,...,p) for $ALG_j$ by fitting $y_{ij}$ and $F_i^k$ (in $TR$)
```

```
          into the regression model
9)      Until no more algorithms in $A$
10)     For each category $C_i$ in $C$
11)       Repeat
12)         Pick one algorithm $ALG_j$ from $A$
13)         Estimate the classification error $\widehat{e}_{ij}$ by fitting $\widehat{\beta}_j^k$ and corresponding $F_i^k$ (in $TU$)
              into the regression model
14)         Calculate estimated classification performance $u_{ij}$ based on $\widehat{e}_{ij}$
15)         Insert $u_{ij}$ into the estimated performance matrix $P$
16)       Until no more algorithms in $A$
17)     End For
18)     Output the final estimated performance matrix $P$
19)     For each category $C_i$ in $C$
20)       Calculate the relative weight factors $\omega_{ij}$ for all algorithms $j$ based on the $u_{ij}$ in $P$
21)     End For
```

Table 6.1: The MUDOF2 algorithm

Steps 1 to 9 in Table 6.1 is exactly the same as the corresponding steps from Steps 1 to 9 shown in Figure 5.2. They aim to estimate a set of betas ($\widehat{\beta}_j^k$), the parameter estimates of the feature characteristics in the regression model, for each individual algorithm. Specifically, each algorithm with optimized parameter settings, is picked from the algorithm pool as indicated in Step 2. The classification errors, and so the preliminary classification performance, of each individual component classification algorithm against each category are obtained by repeating Steps 3 to 7. By fitting all classification errors of the categories, and their corresponding set of document feature characteristics in the training set, into the regression model as shown in Step 8, n sets of estimated betas are obtained after Step 9.

A matrix of the estimated classification performance of the involved algorithms are made from Steps 10 to 18. In Step 12, one algorithm with the same optimized parameter settings as in Step 2, is picked from the algorithm pool. The corresponding set of betas of the selected algorithm, together with the feature characteristics of a category in the tuning set, will be fitted into the regression model in Step 13, to give the estimated classification errors of the algorithm on the category. In Step 14, the predicted errors are converted to estimated classification performanace, which is then inserted into the performance matrix as indicated in Step 15. Similar steps are repeated for each category. The final estimated classification performance matrix is given as the output as shown in Step 18. The estimated classification performance in the matrix is then used for determining the relative weight factors for the component classification algorithms in the remaining steps.

From Steps 19 to 21, the relative weight factors for the component classification algorithms are calculated on category basis. The adopted weighting strategy is the Weighting Strategy Based On Utility Measure as proposed in Section 4.2.2. Under the strategy, those algorithms believed to demonstrate better classification performance during on-line classification will be associated a larger value of the weight factor proportionally. A larger weight factor for a classification algorithm increases the contribution of the algorithm's decisions towards the final combined confidence scores during the linear combination approach.

After calculating the relative weight factors for each component classification algorithms, the approach proceeds by following the same framework as Linear Combination proposed in Section 4.2. Particularly, each component classification algorithm constructs a classifier for a certain category. Confidence scores showing the degree of relevance of each new document to the category is produced by each classifier. The scores are combined under our proposed framework Linear Combination with the use of the determined weight factors. Based on the final combined score, the document is determined to belong to a category if the combined score is larger than a threshold value.

Chapter 7

Experimental Setup

We have implemented our proposed meta-learning approaches, and extensive experiments have been conducted to verify their performance. In this chapter, the details of our experimental setup is given.

7.1 Document Collection

The Reuters-21578 collection contains Reuters newswire articles in 1987. The documents were assembled and labeled with categories by experts from Reuters. There are 21,578 documents in this collection. Each document has been assigned to categories related to financial topics. Some categories appear in many documents while some categories appear in very few documents. The collection is divided into a training document collection and a testing document collection according to the "ModApte" split[1] commonly

[1] The "ModApte" split results in a total of 12,902 documents to be used in the experiments. The remaining 8,676 documents are not used as they have not been classified by human indexer.

used as a benchmark data [7, 38, 48], and we select the 90 categories which have at least one document in both the training set and the testing set. The split results in a total of 9,603 training documents and 3,299 testing documents [2]. As our proposed meta-learning models require a tuning set, we further divided the training collection into training set of 6,000 documents and 3,603 tuning documents. For each category, we used the training document collection to learn a classification scheme. Tuning documents are used for obtaining the preliminary classification performances, which are fitted into our proposed meta-learning models to determine the ways to combine the classification evidence of different component classification algorithms. For example, based on the obtained preliminary performance, the weight factors of different classification algorithms are determined under our Linear Combination approach. Also, based on the obtained preliminary performance, we can calculate the classification errors. By combining the classification errors and categorical document feature characteristics, the parameter estimates for the multivariate regression model can be found and the classification errors can be estimated under MUDOF and MUDOF2. To evaluate the effectiveness of the learned scheme, we used the scheme to classify documents in the testing document collection and compared the result with the manual classification.

The OHSUMED collection is a bibliographical document collection developed by Hersh and his colleagues at the Oregon Health Sciences University.

[2] Other studies may refine the Reuters-21578 corpus by further eliminating those documents that do not belong to those 90 categories, resulting in 7,769 training documents and 3,019 testing documents.

We used 50,216 documents in 1991 which have abstracts. There are total 14,626 distinct main headings appeared in the OHSUMED records. In our experiment, we chose the set of 119 MeSH categories from the heart disease categories. These 119 MeSH heart disease categories were extracted by Yang from the April 1994 (5th Ed.) UMLS CD-ROM, distributed by the National Library of Medicine. The document collection is split, resulting in a total of 38,478 training documents and a total of 11,738 testing documents. Similar to the Reuters-21578 corpus, the training documents in the OHSUMED collection are further split into one training set and one tuning set. The training set contains 33,478 pieces of documents which are used for learning classification schemes for the categories. The tuning document set contains 5,000 pieces of documents for obtaining the preliminary classification performances. The remaining testing documents are used for evaluating the meta-learning models. The OHSUMED corpus is difficult to learn for a good classifier since the documents are very noisy.

7.2 Evaluation Metric

To measure the performance, two common evaluation metrics are used, namely the micro-averaged recall and precision break-even point measure (MBE) and the macro-averaged recall and precision break-even point measure (ABE). These evaluation metrics have been widely used in text categorization experiments [7, 14, 29].

For a particular category, the effectiveness of the classification can be illustrated by a contingency table as follows [26]:

	Expert Says Yes	Expert Says No	
System Says Yes	q	r	$q+r$
System Says No	s	t	$s+t$
	$q+s$	$r+t$	$q+r+s+t$

where q is the number of documents belonging to the category and assigned to the category (true positive); r is the number of documents not belonging to the category but assigned to the category (false positive); s is the number of documents belonging to the category but not assigned to the category (false negative); t is the number of documents not belonging to the category and not assigned to the category (true negative). Some common effectiveness measures can then be defined in terms of these values:

$$(recall)\ R = \frac{q}{(q+s)}$$

$$(precision)\ P = \frac{q}{(q+r)}$$

Recall is the proportion of documents belonging to the category that the system successfully assigns to the category. Precision is the proportion of documents assigned to the category by the system that really belong to the category. An ideal classification system would have both recall and precision equal to 1. However, perfect recall can be achieved by a system that puts every document in the category, while perfect precision can be achieved by a system that puts no documents in the category. Therefore, just using either recall or precision does not provide a fair evaluation to system.

In micro-averaged recall and precision break-even point (MBE) measure, the total number of false positive, false negative, true positive, and true negative are computed across all categories. These totals are used to compute the micro-recall and micro-precision. Then we use the interpolation to find the break-even point where the micro-recall and micro-precision are equal.

In macro-averaged recall and precision break-even point (ABE) measure, the number of false positive, false negative, true positive, and true negative are computed for each category. Based on these totals, the recall and precision break-even point is calculated for each individual category. Then simple average of all those break-even points is taken across all the categories to obtain the final score.

In order to evaluate the meta-models in different perspectives, we adopt the following aspects of measure based on MBE and ABE:

1. ***All MBE***: Aspect of measure computes the MBE for all of the categories in a document collection.

2. ***All ABE***: Aspect of measure computes the ABE for all of the categories in a document collection.

3. ***Top 10 ABE***: Aspect of measure computes the ABE for the ten most frequent categories, which are those categories with top-ten number of positive training documents.

4. ***Other ABE***: Aspect of measure computes the ABE for the remaining categories (less frequent categories) other than the top-ten frequent categories.

Both *All MBE* and *Top 10 ABE* measures favour more frequent categories, which are those categories with more available training examples.

While *All ABE* and *Other ABE* measures are more capable of revealing the classification performance for less frequent categories, which are those categories with less available training examples.

7.3 Component Classification Algorithms

Six component classification algorithms have been used in our meta-model approaches. They are Rocchio, WH, KNN, SVM, GISR and GISW, with optimized parameter settings. Each of these algorithms exhibits certain distinctive nature: Rocchio and WH are linear classifiers, KNN is an instance-based learning algorithm, SVM is based on Structural Risk Minimization Principle [42] and both GISR and GISW [21] are based on the generalized instance approach.

Different values of parameters have been tried on each algorithm to ensure that the most optimized parameters setting is used for each component classification algorithms. For the Reuters-21578 corpus, the values of η tried for the Rocchio algorithm included 0.0, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95, 1.0. The values of k tried for the KNN algorithm in this corpus were 30, 50, 70, 100, 150, 300, 500, 550, 600. For the OHSUMED corpus, the values of η tried for the Rocchio algorithm included 0.0, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0. The values of k tried for the KNN algorithm included 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800.

7.4 Categorical Document Feature Characteristics for MUDOF and MUDOF2

In MUDOF, eight document feature characteristics are used in our regression model as independent variables:

1. ***PosTr***: The number of positive training examples of a category.
2. ***PosTu***: The number of positive tuning examples of a category.
3. ***AvgDocLen***: The average document length of a category. Document length refers to the number of indexed terms within a document. The average is taken across all the positive examples of a category.
4. ***AvgTermVal***: The average term weight of documents across a category. Average term weight is taken for individual documents first. Then, the average is taken across all the positive examples of a category.
5. ***AvgMaxTermVal***: The average maximum term weight of documents across a category. Maximum term weight of individual documents are summed, and the average is taken across all the positive examples of a category.
6. ***AvgMinTermVal***: The average minimum term weight of documents across a category. Minimum term weight of individual documents are summed, and the average is taken across all the positive examples of a category.
7. ***AvgTermThre***: The average number of terms above a term weight threshold. The term weight threshold is optimized and set globally. Based on the preset threshold, the number of terms with term weight above the threshold within a category are summed. The average is then taken across all the positive examples of the category.
8. ***AvgTopInfoGain***: The average information gain of the top m terms of a category. The information gain of each individual term is calculated for each category and ranked. The average is then taken across the top m terms with highest information

gain.

$$\begin{aligned} G(t) = &- \sum_{i=1}^{m} P_r(c_i) \log P_r(c_i) + \\ &P_r(t) \sum_{i=1}^{m} P_r(c_i|t) \log P_r(c_i|t) + \\ &P_r(\bar{t}) \sum_{i=1}^{m} P_r(c_i|\bar{t}) \log P_r(c_i|\bar{t}) \end{aligned} \tag{7.1}$$

Two sets of normalized feature characteristics are collected separately from the training set and the tuning set. As illustrated in Step 8 and Step 13 in Figure 5.2, the feature characteristics from these two data sets serve different purposes in MUDOF and MUDOF2: feature characteristics from training set are used for obtaining the preliminary classification performance based on which parameters are estimated, while feature characteristics from tuning set are used for predicting classification errors, base on which algorithms are recommended for each category.

Chapter 8

Experimental Results and Analysis

Extensive experiments have been conducted on two real-world document collection, namely, the Reuters-21578 corpus and the OHSUMED collection. The experiments are set up as mentioned in Chapter 7. The results are summarized and evaluated based on the MBE and ABE measures as described in Chapter 7.2. More details of experimental results are shown in the Appendix A for the Reuters-21578 corpus, and in the Appendix B for the OHSUMED corpus.

8.1 Performance of Linear Combination Approach

Under the Linear Combination Approach, we have proposed three different weighting strategies in order to combine the classification decisions of different classification algorithms linearly. The three strategies are Equal Weighting Strategy (LC1), Weighting Strategy Based On Utility Measure (LC2) and Weighting Strategy Based On Document Rank (LC3). Experiments are run by using these three weight strategies.

Table 8.1 shows the macro-averaged recall and precision break-even point measure for the ten most frequent categories using the Reuters-21578 document corpus. Among the three strategies, the Weighting Strategy Based On Utility Measure is the best. Also, both Equal Weighting Strategy and Weighting Strategy Based On Document Rank outperform all other component classification algorithms.

Table 8.2 summarizes the performance of the Linear Combination approach with different weighting strategies in different perspectives of measure. Classification performances are compared between the three strategies and the individual component classification algorithms. Based on Table 8.2, the corresponding percentage improvement of the Linear Combination approach over existing component algorithms is shown in Table 8.3.

Table 8.3 shows the percentage improvement of the Linear Combination approach over existing classification algorithms. Among all the component classification algorithms, the classification improvement of Linear Combination approach over Rocchio is the largest, with over 10% in all aspects of measure. When compared with the classification performance achieved by SVM and KNN, the Linear Combination approach also shows satisfactory classification improvement. The table shows that the overall improvement for more frequent categories (*All MBE* and *Top 10 ABE*) is more significant than that of less frequent categories. When compared with WH and GISW, Linear Combination approach demonstrates little inferior performance for less frequent categories. This may be attributed to the fact that the relative contributions of certain component classification algorithms are not truly affected during the classification for the Reuters-21578 document collection.

CAT	RO	WH	KNN	LC1	LC2	LC3
acq	0.829	0.870	0.859	0.947	0.947	0.949
corn	0.614	0.867	0.690	0.846	0.853	0.867
crude	0.793	0.853	0.823	0.860	0.860	0.860
earn	0.956	0.969	0.956	0.979	0.979	0.978
grain	0.803	0.887	0.820	0.896	0.897	0.896
interest	0.702	0.749	0.712	0.790	0.796	0.794
money-fx	0.582	0.718	0.674	0.763	0.764	0.758
ship	0.800	0.860	0.800	0.884	0.883	0.883
trade	0.732	0.763	0.740	0.792	0.797	0.784
wheat	0.713	0.839	0.727	0.825	0.839	0.825
Top 10 ABE	0.752	0.838	0.780	0.858	0.862	0.859
CAT	SVM	GISR	GISW	LC1	LC2	LC3
acq	0.931	0.932	0.909	0.947	0.947	0.949
corn	0.832	0.867	0.885	0.846	0.853	0.867
crude	0.871	0.813	0.869	0.860	0.860	0.860
earn	0.980	0.959	0.962	0.979	0.979	0.978
grain	0.917	0.804	0.910	0.896	0.897	0.896
interest	0.619	0.758	0.745	0.790	0.796	0.794
money-fx	0.717	0.681	0.756	0.763	0.764	0.758
ship	0.845	0.825	0.872	0.884	0.883	0.883
trade	0.715	0.714	0.788	0.792	0.797	0.784
wheat	0.820	0.825	0.875	0.825	0.839	0.825
Top 10 ABE	0.825	0.818	0.857	0.858	0.862	0.859

Table 8.1: Comparison of Linear Combination approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus.

Table 8.4 shows the macro-averaged recall and precision break-even point measure for the ten most frequent categories using the OHSUMED document

MEASURE	RO	WH	KNN	LC1	LC2	LC3
All MBE	0.776	0.820	0.802	0.860	0.861	0.858
All ABE	0.578	0.649	0.607	0.647	0.649	0.644
Top 10 ABE	0.752	0.838	0.780	0.858	0.862	0.859
Other ABE	0.556	0.625	0.585	0.621	0.622	0.617
MEASURE	SVM	GISR	GISW	LC1	LC2	LC3
All MBE	0.841	0.830	0.845	0.860	0.861	0.858
All ABE	0.640	0.625	0.655	0.647	0.649	0.644
Top 10 ABE	0.825	0.818	0.857	0.858	0.862	0.859
Other ABE	0.617	0.601	0.630	0.621	0.622	0.617

Table 8.2: Comparison of classification performance of Linear Combination approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus.

	RO			WH		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	10.825	10.954	10.567	4.878	5.000	4.634
All ABE	11.930	12.284	11.419	-0.254	0.062	-0.709
Top 10 ABE	14.096	14.628	14.229	2.387	2.864	2.506
Other ABE	11.691	11.871	10.971	-0.640	-0.480	-1.280
	KNN			SVM		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	7.232	7.357	6.983	2.259	2.378	2.021
All ABE	6.621	6.958	6.134	1.025	1.345	0.564
Top 10 ABE	10.000	10.513	10.128	4.000	4.485	4.121
Other ABE	6.154	6.325	5.470	0.648	0.810	0.000
	GISR			GISW		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	3.614	3.735	3.373	1.775	1.893	1.538
All ABE	3.452	3.779	2.980	-1.270	-0.958	-1.721
Top 10 ABE	4.890	5.379	5.012	0.117	0.583	0.233
Other ABE	3.328	3.494	2.662	-1.429	-1.270	-2.063

Table 8.3: Percentage improvement of the Linear Combination approach over existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus.

corpus. Among the three strategies, the Equal Weighting Strategy shows the best performance. Also, Weighting Strategy Based On Utility Measure and Weighting Strategy Based On Document Rank outperform all other component classification algorithms.

Table 8.5 summarizes the performance of the Linear Combination approach with different weighting strategies in different perspectives. Performances are compared between the three strategies and the individual component classification algorithms, and the corresponding percentage improvement of the Linear Combination approach is shown in Table 8.6.

Table 8.6 shows the percentage improvement of the Linear Combination approach over existing classification algorithms. Similar to the case of using the Reuters document collection, among all the component classification algorithms, the classification improvement of the Linear Combination approach over Rocchio is the largest, with over 10% in all aspects of measure. When compared with the classification performance achieved by SVM and KNN, the Linear Combination approach also demonstrates good classification improvement. In general, when compared with the case of using the Reuters corpus, the Linear Combination approach achieves much significant improvement for the OHSUMED corpus.

8.2 Performance of the MUDOF Approach

Our MUDOF approach aims to predict a classification error of a certain classification algorithm for a category based on the the categorical document feature characteristics with the use of a regression model. The learned re-

CAT	RO	WH	KNN	LC1	LC2	LC3
Angina Pectoris	0.344	0.536	0.454	0.485	0.490	0.490
Arrhythmia	0.541	0.575	0.536	0.580	0.572	0.595
Coronary Arteriosclerosis	0.267	0.356	0.218	0.400	0.400	0.385
Coronary Disease	0.466	0.540	0.552	0.579	0.572	0.582
Heart Arrest	0.638	0.609	0.580	0.638	0.629	0.638
Heart Defects, Congenital	0.493	0.678	0.543	0.667	0.678	0.657
Heart Diseases	0.225	0.222	0.139	0.310	0.310	0.310
Heart Failure, Congestive	0.436	0.602	0.552	0.603	0.586	0.586
Myocardial Infarction	0.781	0.811	0.789	0.812	0.818	0.809
Tachycardia	0.684	0.582	0.684	0.684	0.684	0.684
Top 10 ABE	0.488	0.551	0.505	0.576	0.574	0.574

CAT	SVM	GISR	GISW	LC1	LC2	LC3
Angina Pectoris	0.449	0.516	0.598	0.485	0.490	0.490
Arrhythmia	0.432	0.584	0.572	0.580	0.572	0.595
Coronary Arteriosclerosis	0.356	0.311	0.445	0.400	0.400	0.385
Coronary Disease	0.502	0.556	0.565	0.579	0.572	0.582
Heart Arrest	0.543	0.638	0.609	0.638	0.629	0.638
Heart Defects, Congenital	0.644	0.534	0.610	0.667	0.678	0.657
Heart Diseases	0.190	0.197	0.222	0.310	0.310	0.310
Heart Failure, Congestive	0.493	0.556	0.602	0.603	0.586	0.586
Myocardial Infarction	0.750	0.832	0.799	0.812	0.818	0.809
Tachycardia	0.608	0.700	0.633	0.684	0.684	0.684
Top 10 ABE	0.497	0.542	0.566	0.576	0.574	0.574

Table 8.4: Comparison of Linear Combination approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus.

MEASURE	RO	WH	KNN	LC1	LC2	LC3
All MBE	0.504	0.552	0.534	0.591	0.582	0.591
All ABE	0.442	0.484	0.466	0.510	0.501	0.511
Top 10 ABE	0.488	0.551	0.505	0.576	0.574	0.574
Other ABE	0.436	0.474	0.460	0.501	0.490	0.502
MEASURE	SVM	GISR	GISW	LC1	LC2	LC3
All MBE	0.539	0.575	0.583	0.591	0.582	0.591
All ABE	0.485	0.496	0.483	0.510	0.501	0.511
Top 10 ABE	0.497	0.542	0.566	0.576	0.574	0.574
Other ABE	0.484	0.490	0.472	0.501	0.490	0.502

Table 8.5: Comparison of classification performance of Linear Combination approach with existing component classification algorithms in different perspectives of measure for the OHSUMED corpus.

	RO			WH		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	17.262	15.476	17.262	7.065	5.435	7.065
All ABE	15.385	13.348	15.611	5.372	3.512	5.579
Top 10 ABE	18.033	17.623	17.623	4.537	4.174	4.174
Other ABE	14.908	12.385	15.138	5.696	3.376	5.907
	KNN			SVM		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	10.674	8.989	10.674	9.647	7.978	9.647
All ABE	9.442	7.511	9.657	5.155	3.299	5.361
Top 10 ABE	14.059	13.663	13.663	15.895	15.493	15.493
Other ABE	8.913	6.522	9.130	3.512	1.240	3.719
	GISR			GISW		
	LC1(%)	LC2(%)	LC3(%)	LC1(%)	LC2(%)	LC3(%)
All MBE	2.783	1.217	2.783	1.372	-0.172	1.372
All ABE	2.823	1.008	3.024	5.590	3.727	5.797
Top 10 ABE	6.273	5.904	5.904	1.767	1.413	1.413
Other ABE	2.245	0.000	2.449	6.144	3.814	6.356

Table 8.6: Percentage improvement of the Linear Combination approach over existing component classification algorithms under different perspectives of measure for the OHSUMED corpus.

lationship is expressed in the form of a set of parameter estimates for each component algorithm. Table 8.7 shows the sets of parameter estimates of the component classification algorithms for the Reuters-21578 corpus. Based on these parameter estimates and the corresponding feature characteristics, on category basis, the estimated classification errors of different algorithms against different categories can be obtained.

Besides of comparing the classification performance of the MUDOF approach against individual classification algorithms, we also set up the ideal combination of algorithms as another benchmark for our MUDOF approach. To set up the ideal combination of algorithms (IDEAL), we manually select the best algorithms for each category according to their real classification performances. Since IDEAL consists of the best algorithms for each category, it represents the theoretical perfect improvement could be achieved under the framework of choosing one algorithm for each category. In short, IDEAL sets an upper bound for the amount of improvement that can be made under MUDOF approach. Table 8.8 shows the classification performance achieved by both MUDOF approach and the IDEAL combination for the ten most frequent categories. Items in bold are those algorithms correctly recommended by MUDOF. Our results, show that MUDOF can identify the ideal algorithms for 56 categories out of the total 90 categories in the Reuters corpus, with an accuracy of over 62%. When neglecting the ten most frequent categories, MUDOF can still identify the ideal algorithms for 50 categories out of the 80 less frequent categories, also with an accuracy over 62%. The results show that MUDOF can demonstrate consistent performance of algorithm recommendation in the Reuters-21578 corpus.

Table 8.9 shows the comparison of performances, under different aspects of measure, between MUDOF and the IDEAL combination over other component algorithms. Based on the utility measures as shown in the table, we look into how much improvement the meta-model MUDOF (M+(%)) has achieved within the improvement bound (I+(%)) set by the ideal combination (IDEAL) in Table 8.10.

In Table 8.10, among all other algorithms, the classification improvement made by either MUDOF (M+(%)) or the IDEAL combination (I+(%)) over Rocchio is the largest, more than 10% on average in most aspects of measure. Improvement made by MUDOF over KNN is also significant, it is more than 5% in all aspects. Our MUDOF approach can even make improvement for less frequent categories in the Reuters corpus. For example, the approach achieves a considerable amount of improvement for less frequent categories over GISW and WH, both of which demonstrate better classification performance than the Linear Combination approach using the same document collection.

Table 8.10 also reveals that the improvement made by MUDOF over individual component algorithms is impressive when considering the improvement bound set by the ideal combination (I+(%)). Improvement achieved by MUDOF within the improvement bound of the ideal combination ($M+/I+(\%)$) is presented in the table. When compared with Rocchio and KNN, the meta-model has attained more than 50% of the improvement bound in all aspects of measure. As for All MBE measure, MUDOF can also achieve from more than 20% to more than 70% of the improvement bound for most of the component algorithms.

Features	RO	WH	KNN	SVM	GISR	GISW
PosTr	14.011	25.408	20.882	24.903	27.232	32.065
PosTu	-6.038	-18.348	-8.390	-11.036	-20.316	-24.198
AvgDocLen	1133.713	1879.174	786.264	1198.236	1099.567	1525.261
AvgMaxTermVal	0.868	5.697	-4.168	-5.171	1.400	3.845
AvgMinTermVal	-24.885	-54.505	-27.211	-28.588	-48.487	-62.657
AvgTermVal	54.972	87.502	36.261	49.658	68.046	81.286
AvgTermThre	-1123.324	-1860.777	-778.078	-1189.399	-1085.458	-1505.405
AvgTopInfoGain	-31.395	-31.682	-41.838	-50.206	-42.711	-39.972
Intercept	-13.674	-24.452	-6.866	-9.712	-16.157	-21.879

Table 8.7: Parameter estimates for categorical document feature characteristics of different algorithms for the Reuters-21578 corpus.

Table 8.12 shows the classification performance achieved by both MUDOF approach and the IDEAL combination for the ten most frequent categories with the OHSUMED document collection. Items in bold are those algorithms recommended by MUDOF correctly. Our results show that the MUDOF can identify the ideal algorithms for 44 categories out of the total 81 categories in the OHSUMED corpus, with an accuracy of over 54%. When neglecting the ten most frequent categories, MUDOF can also identify the ideal algorithms for 42 categories out of the 71 less frequent categories, with an accuracy over 59%. Unlike the case of using the Reuters corpus, the performance difference for frequent and less frequent categories leads to the fact that MUDOF can demonstrate more consistent performance of algorithm recommendation for less frequent categories in the OHSUMED corpus.

Table 8.13 shows the comparison of performances, under different aspects of measure, between MUDOF and the IDEAL combination over other component algorithms. Table 8.14 shows how much improvement MUDOF (M+(%)) has achieved within the improvement bound (I+(%)) set by the

CAT	RO	WH	KNN	MUDOF		IDEAL
acq	0.829	0.870	0.859	0.909	GISW	GISR
corn	0.614	0.867	0.690	0.885	**GISW**	GISW
crude	0.793	0.853	0.823	0.853	WH	SVM
earn	0.956	0.969	0.956	0.980	**SVM**	SVM
grain	0.803	0.887	0.820	0.910	GISW	SVM
interest	0.702	0.749	0.712	0.745	GISW	GISR
money-fx	0.582	0.718	0.674	0.756	**GISW**	GISW
ship	0.800	0.860	0.800	0.872	**GISW**	GISW
trade	0.732	0.763	0.740	0.788	**GISW**	GISW
wheat	0.713	0.839	0.727	0.875	**GISW**	GISW
Top 10 ABE	0.752	0.838	0.780	0.857		0.863

CAT	SVM	GISR	GISW	MUDOF		IDEAL
acq	0.931	0.932	0.909	0.909	GISW	GISR
corn	0.832	0.867	0.885	0.885	**GISW**	GISW
crude	0.871	0.813	0.869	0.853	WH	SVM
earn	0.980	0.959	0.962	0.980	**SVM**	SVM
grain	0.917	0.804	0.910	0.910	GISW	SVM
interest	0.619	0.758	0.745	0.745	GISW	GISR
money-fx	0.717	0.681	0.756	0.756	**GISW**	GISW
ship	0.845	0.825	0.872	0.872	**GISW**	GISW
trade	0.715	0.714	0.788	0.788	**GISW**	GISW
wheat	0.820	0.825	0.875	0.875	**GISW**	GISW
Top 10 ABE	0.825	0.818	0.857	0.857		0.863

Table 8.8: Comparison of the MUDOF approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus.

MEASURE	RO	WH	KNN	MUDOF	IDEAL
All MBE	0.776	0.820	0.802	0.847	0.868
All ABE	0.578	0.649	0.607	0.659	0.692
Top 10 ABE	0.752	0.838	0.780	0.857	0.863
Other ABE	0.556	0.625	0.585	0.634	0.670

MEASURE	SVM	GISR	GISW	MUDOF	IDEAL
All MBE	0.841	0.830	0.845	0.847	0.868
All ABE	0.640	0.625	0.655	0.659	0.692
Top 10 ABE	0.825	0.818	0.857	0.857	0.863
Other ABE	0.617	0.601	0.630	0.634	0.670

Table 8.9: Comparison of classification performance of the MUDOF approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus.

	RO			WH		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	9.149	11.856	77.168	3.293	5.854	56.252
All ABE	14.014	19.723	71.054	1.603	6.691	23.958
Top 10 ABE	13.963	14.761	94.594	2.267	2.983	33.881
Other ABE	14.029	20.504	68.421	1.440	7.200	21.521

	KNN			SVM		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	5.611	8.229	68.186	0.713	3.210	22.212
All ABE	8.606	14.045	61.274	2.906	8.059	36.059
Top 10 ABE	9.872	10.641	92.773	3.879	4.606	84.216
Other ABE	8.376	14.530	57.646	2.755	8.590	32.072

	GISR			GISW		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	2.048	4.578	44.736	0.237	2.722	8.707
All ABE	5.378	10.655	50.474	0.568	5.604	10.136
Top 10 ABE	4.768	5.501	86.675	0.000	0.700	0.000
Other ABE	5.491	11.481	47.827	0.635	6.349	10.002

Table 8.10: Improvement of classification performances of MUDOF (M+(%)) and the ideal combination (I+(%)) over individual classification algorithms, and improvement achieved by MUDOF within the improvement bound set by the ideal combination (M+/I+(%)) for the Reuters-21578 corpus.

ideal combination.

Table 8.14 shows a similar trend as the case of using the Reuters document collection that, among all other algorithms, the classification improvement made by either MUDOF (M+(%)) or the ideal combination (I+(%)) over Rocchio is the largest, with more than 10% improvement on average, in most aspects of measure. Improvement made by the MUDOF over KNN is also significant, it is more than 5% in all aspects. Similar to the case of using the Reuters document collection, our MUDOF approach continues to make improvement for less frequent categories. For example, the approach achieves much more significant improvement for less frequent categories (*All ABE* and *Other ABE*) over GISW and WH, which demonstrate better classification performance when compared with the Linear Combination approach using the same document collection.

Table 8.14 also reveals that the improvement made by MUDOF over individual component algorithms is also satisfactory when considering the improvement bound set by the ideal combination (I+(%)). Improvement achieved by MUDOF within the improvement bound of the ideal combination ($M + /I + (\%)$) is presented in the table. When compared with Rocchio, KNN and SVM, the meta-model has attained from more than 10% to over 50% of the improvement bound for all aspects. As mentioned before, MUDOF demonstrates better performance for less frequent categories than the frequent categories in the OHSUMED corpus. Furthermore, the measure of All MBE is largely affected by the performance of the more frequent categories. As a result, MUDOF demonstrates less efficiently under the measures of All MBE and Top 10 ABE measures when compared against certain component algorithms.

Features	RO	WH	KNN	SVM	GISR	GISW
PosTr	897.924	767.279	1024.344	781.367	975.184	1392.322
PosTu	-375.800	-262.152	-350.087	-275.610	-390.509	-517.932
AvgDocLen	-10.998	-8.059	-10.500	-12.998	-13.366	-4.635
AvgMaxTermVal	-17.268	-10.773	-17.242	-2.479	-14.526	-11.238
AvgMinTermVal	197.000	96.996	237.458	135.631	173.697	143.398
AvgTermVal	-109.754	-76.440	-119.414	-108.587	-114.186	-76.139
AvgTermThre	-22.035	-3.982	-23.841	10.275	-14.462	-9.987
AvgTopInfoGain	-654.789	-736.868	-851.010	-718.994	-755.422	-1162.992
Intercept	28.493	17.962	28.902	14.659	27.097	17.118

Table 8.11: Parameter estimates for categorical document feature characteristics of different algorithms for the OHSUMED corpus.

8.3 Performance of MUDOF2 Approach

MUDOF2 aims to derive the relative weight factors for each component classification algorithm with consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category. Table 8.15 shows the classification performance of MUDOF2 for the ten most frequent categories. It shows that MUDOF2 achieves a better classification performance than any individual component classifiers for the ten most frequent categories.

Table 8.16 summarizes and compares the classification performance of MUDOF2 and Linear Combination approach using the Weighting Strategy Based On Utility Measure, over the component algorithms. MUDOF2

CAT	RO	WH	KNN	MUDOF		IDEAL
Angina Pectoris	0.344	0.536	0.454	0.598	**GISW**	GISW
Arrhythmia	0.541	0.575	0.536	0.536	KNN	GISR
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	SVM	GISW
Coronary Disease	0.466	0.540	0.552	0.502	SVM	GISW
Heart Arrest	0.638	0.609	0.580	0.580	KNN	GISR/RO
Heart Defects, Congenital	0.493	0.678	0.543	0.644	SVM	WH
Heart Diseases	0.225	0.222	0.139	0.190	SVM	RO
Heart Failure, Congestive	0.436	0.602	0.552	0.602	**GISW**	GISW/WH
Myocardial Infarction	0.781	0.811	0.789	0.799	GISW	GISR
Tachycardia	0.684	0.582	0.684	0.608	SVM	KNN/RO
Top 10 ABE	0.488	0.551	0.505	0.542		0.585

CAT	SVM	GISR	GISW	MUDOF		IDEAL
Angina Pectoris	0.449	0.516	0.598	0.598	**GISW**	GISW
Arrhythmia	0.432	0.584	0.572	0.536	KNN	GISR
Coronary Arteriosclerosis	0.356	0.311	0.445	0.356	SVM	GISW
Coronary Disease	0.502	0.556	0.565	0.502	SVM	GISW
Heart Arrest	0.543	0.638	0.609	0.580	KNN	GISR/RO
Heart Defects, Congenital	0.644	0.534	0.610	0.644	SVM	WH
Heart Diseases	0.190	0.197	0.222	0.190	SVM	RO
Heart Failure, Congestive	0.493	0.556	0.602	0.602	**GISW**	GISW/WH
Myocardial Infarction	0.750	0.832	0.799	0.799	GISW	GISR
Tachycardia	0.608	0.700	0.633	0.608	SVM	KNN/RO
Top 10 ABE	0.497	0.542	0.566	0.542		0.585

Table 8.12: Comparison of the MUDOF approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus.

MEASURE	RO	WH	KNN	MUDOF	IDEAL
All MBE	0.504	0.552	0.534	0.561	0.607
All ABE	0.442	0.484	0.466	0.501	0.560
Top 10 ABE	0.488	0.551	0.505	0.542	0.585
Other ABE	0.436	0.474	0.460	0.495	0.557
MEASURE	SVM	GISR	GISW	MUDOF	IDEAL
All MBE	0.539	0.575	0.583	0.561	0.607
All ABE	0.485	0.496	0.483	0.501	0.560
Top 10 ABE	0.497	0.542	0.566	0.542	0.585
Other ABE	0.484	0.490	0.472	0.495	0.557

Table 8.13: Comparison of classification performance of the MUDOF approach with existing component classification algorithms under different perspectives of measure for the OHSUMED corpus.

	RO			WH		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	11.310	20.437	55.340	1.630	9.964	16.364
All ABE	13.348	26.697	50.000	3.512	15.702	22.368
Top 10 ABE	11.066	19.877	55.670	-1.633	6.171	-26.471
Other ABE	13.532	27.752	48.760	4.430	17.511	25.301
	KNN			SVM		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	5.056	13.670	36.986	4.082	12.616	32.353
All ABE	7.511	20.172	37.234	3.299	15.464	21.333
Top 10 ABE	7.327	15.842	46.250	9.054	17.706	51.136
Other ABE	7.609	21.087	36.082	2.273	15.083	15.068
	GISR			GISW		
	M+(%)	I+(%)	M+/I+(%)	M+(%)	I+(%)	M+/I+(%)
All MBE	-2.435	5.565	-43.750	-3.774	4.117	-91.667
All ABE	1.008	12.903	7.813	3.727	15.942	23.377
Top 10 ABE	0.000	7.934	0.000	-4.240	3.357	-126.316
Other ABE	1.020	13.673	7.463	4.873	18.008	27.059

Table 8.14: Improvement of classification performances of MUDOF (M+(%)) and the ideal combination (I+(%)) over individual classification algorithms, and improvement achieved by MUDOF within the improvement bound set by the ideal combination (M+/I+(%)) for the OHSUMED corpus.

achieves an overall better classification performance over all existing component classification algorithms in most aspects of measure. Also, it improves the Linear Combination approach for most of the perspectives of evaluation.

Table 8.17 compares the percentage improvement made by MUDOF2 and Linear Combination approach. Results show that, under various aspects of evaluation, MUDOF2 demonstrates improvement of classification performance over the component algorithms, except GISW which performs marginally better than MUDOF2 under ABE measures. The improvement of classification performance achieved by MUDOF2 is most significant when compared against Rocchio, followed by KNN and SVM. There is more than 10% of improvement under all measures when compared against Rocchio, and more than 7% when compared with KNN. When compared with the robust SVM, MUDOF2 can still demonstrate from more than 1% to 4% improvement. The improvement of MUDOF2 over individual algorithms is more significant and unique for the results of OHSUMED corpus.

Moreover, Table 8.17 also shows that MUDOF2 can improve Linear Combination approach in various extents under most aspects of measure. Particularly, the absolute increase in percentage improvement made by MUDOF2 over Linear Combination is consistently more than 0.8% under the Other ABE measure when compared with all component algorithms. By using MUDOF2, the incremental improvement percentage achieved over the improvement made by LC2 against SVM, WH and GISW is very significant. The incremental improvement ranges from more than 60% to nearly 10 times of the improvement made by LC2 for the less frequent categories (*All ABE* and *Other ABE*). Since the measure under the Top 10 ABE is limited to just a small proportion of categories of the whole document collection, and therefore, its slightly inferior performances can be offset by the overall increased performances in all other aspects of evaluation, which involves the major proportion of the total number of categories. As a result, by adding up the combined incremental improvement made by MUDOF2 over Linear Combination, it can be observed that incorporating Linear Combination into MUDOF does help to improve the overall classification performance. Much more significant improvement by MUDOF2 over Linear Combination is demonstrated by using the OHSUMED document collection.

CAT	RO	WH	KNN	SVM	GISR	GISW	MUDOF2
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.947
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.867
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.863
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.979
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.908
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.785
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.756
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.878
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.778
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.847
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.861

Table 8.15: Comparison of the MUDOF2 approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the Reuters-21578 corpus.

By using the OHSUMED document collection, more encouraging results are obtained. Table 8.18 shows the classification performance obtained by MUDOF2 over other component classification algorithms. The overall results show that MUDOF2 outperforms all other component algorithms for the ten

MEASURE	RO	WH	KNN	MUDOF2	LC2
All MBE	0.776	0.820	0.802	0.863	0.861
All ABE	0.578	0.649	0.607	0.653	0.649
Top 10 ABE	0.752	0.838	0.780	0.861	0.862
Other ABE	0.556	0.625	0.585	0.627	0.622
MEASURE	SVM	GISR	GISW	MUDOF2	LC2
All MBE	0.841	0.830	0.845	0.863	0.861
All ABE	0.640	0.625	0.655	0.653	0.649
Top 10 ABE	0.825	0.818	0.857	0.861	0.862
Other ABE	0.617	0.601	0.630	0.627	0.622

Table 8.16: Comparison of classification performance of the MUDOF2 approach with existing component classification algorithms under different perspectives of measure for the Reuters-21578 corpus.

	RO		WH	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	11.211	10.954	5.244	5.000
All ABE	12.976	12.284	0.678	0.062
Top 10 ABE	14.495	14.628	2.745	2.864
Other ABE	12.770	11.871	0.320	-0.480
	KNN		SVM	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	7.606	7.357	2.616	2.378
All ABE	7.618	6.958	1.969	1.345
Top 10 ABE	10.385	10.513	4.364	4.485
Other ABE	7.179	6.325	1.621	0.810
	GISR		GISW	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	3.976	3.735	2.130	1.893
All ABE	4.419	3.779	-0.348	-0.958
Top 10 ABE	5.257	5.379	0.467	0.583
Other ABE	4.326	3.494	-0.476	-1.270

Table 8.17: Improvement of classification performances of MUDOF2 (M2+(%)) and the Linear Combination approach with Weighting Strategy based on Utility Measure (LC2+(%)) over individual classification algorithms for the Reuters-21578 corpus.

most frequent categories.

Table 8.19 compares the performance obtained by MUDOF2 and Linear Combination under different aspects of measure. Not only outperforming other component classification algorithms, the results even demonstrate much more significant improvement over Linear Combination approach in all aspects, including the Top 10 ABE measure.

Table 8.20 compares the percentage improvement achieved by MUDOF2 and Linear Combination approach. The results uniquely confirms that the combination of MUDOF and Linear Combination approach can not just improve the classification performance over individual component algorithms, but also outperforms the Linear Combination approach in all aspects of measure. When compared with the results obtained by using the Reuters-21578 document collection, the results for the OHSUMED corpus demonstrate more unique and significant improvement over component algorithms, including GISW. Similar to the results for the Reuters-21578 corpus, improvement over Rocchio is again the largest for the OHSUMED collection, having more than 18% improvement under all aspects of measure. When compared with KNN, the improvement is over 116% to more than 17%.

Furthermore, based on Table 8.20, we find that the absolute increase in percentage improvement for MUDOF2 over Linear Combination is the largest for the Other ABE measure, similar to the case of using the Reuters corpus, with more than 5% increase when compared with all component algorithms. When considering the incremental improvement percentage obtained by MUDOF2 over Linear Combination, the results are more significant and unique. Particularly, the incremental improvement of percentage improvement made by MUDOF2 over Linear Combination against SVM under the Other ABE measure is over 400%, while that made by MUDOF2 over GISW under the All MBE measure is nearly 10 times. When compared with GISR, the incremental improvement made over Linear Combination approach under All ABE measure is over 440%. When compared with KNN, the incremental improvement achieved by MUDOF2 ranges from more than 11% to more than 83% in different aspects of evaluation. With the unique improvement obtained for the OHSUMED corpus, we can conclude again that MUDOF2 can not just improve the individual component algorithms, but also improve the Linear Combination approach.

CAT	RO	WH	KNN	SVM	GISR	GISW	MUDOF2
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.495
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.580
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.400
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.582
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.638
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.667
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.357
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.602
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.812
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.684
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.582

Table 8.18: Comparison of the MUDOF2 approach with existing component classification algorithms based on macro-averaged recall and precision break-even point measures of the ten most frequent categories in the OHSUMED corpus.

MEASURE	RO	WH	KNN	MUDOF2	LC2
All MBE	0.504	0.552	0.534	0.597	0.582
All ABE	0.442	0.484	0.466	0.523	0.501
Top 10 ABE	0.488	0.551	0.505	0.582	0.574
Other ABE	0.436	0.474	0.460	0.515	0.490
MEASURE	SVM	GISR	GISW	MUDOF2	LC2
All MBE	0.539	0.575	0.583	0.597	0.582
All ABE	0.485	0.496	0.483	0.523	0.501
Top 10 ABE	0.497	0.542	0.566	0.582	0.574
Other ABE	0.484	0.490	0.472	0.515	0.490

Table 8.19: Comparison of classification performance of the MUDOF2 approach with existing component classification algorithms under different perspectives of measure for the OHSUMED corpus.

	RO		WH	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	18.452	15.476	8.152	5.435
All ABE	18.326	13.348	8.058	3.512
Top 10 ABE	19.262	17.623	5.626	4.174
Other ABE	18.119	12.385	8.650	3.376
	KNN		SVM	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	11.798	8.989	10.761	7.978
All ABE	12.232	7.511	7.835	3.299
Top 10 ABE	15.248	13.663	17.103	15.493
Other ABE	11.957	6.522	6.405	1.240
	GISR		GISW	
	M2+(%)	LC2+(%)	M2+(%)	LC2+(%)
All MBE	3.826	1.217	2.401	-0.172
All ABE	5.444	1.008	8.282	3.727
Top 10 ABE	7.380	5.904	2.827	1.413
Other ABE	5.102	0.000	9.110	3.814

Table 8.20: Improvement of classification performances of MUDOF2 (M2+(%)) and the Linear Combination approach with Weighting Strategy based on Utility Measure (LC2+(%)) over individual classification algorithms for the OHSUMED corpus.

Chapter 9

Conclusions and Future Work

9.1 Conclusions

We have conducted research on new approaches for meta-learning models of automatic textual document categorization. We investigate the Linear Combination (LC) approach by distilling the characteristic of how we estimate the relative merit of each component algorithm for different categories. Under the linear combination framework, we propose three different weighting strategies, which are used for determining the relative contribution of the component algorithms towards the final classification decisions. Extensive experiments have been conducted on two large-scale, real-world document corpora, namely the Reuters-21578 document collection and the OHSUMED document collection. Results show that the approach demonstrates improvement of classification performance under different perspectives of evaluation.

The Linear Combination approach makes use of limited knowledge in the training document set. To address this limitation, we propose a novel meta-model approach, called Meta-learning Using Document Feature characteristics (MUDOF). MUDOF makes use of category specific document feature characteristics and multivariate regression analysis. By learning the relationship between categorical document feature characteristics and the classification errors of different algorithms, classification errors of each component algorithm are predicted. Based on the predicted errors, the approach is able to recommend the most ideal component algorithms for each category. Experimental results show that the approach can achieve very satisfactory accuracy of prediction for the ideal algorithms, and also confirm that capturing categorical document feature characteristics helps improve the overall classification performances over other existing algorithms.

By incorporating MUDOF into Linear Combination approach, we further propose the third meta-learning approach, MUDOF2. Different from the Linear Combination approach, MUDOF2 can derive the relative weight factors for each component classification algorithm with proper consideration of categorical document feature characteristics. By capturing the document feature characteristics for the determination of weight factors, the relative contribution of each component classification algorithm can be truly reflected with the more comprehensive knowledge of the nature of a category. Extensive experiments have been conducted on the Reuters-21578 collection and the OHSUMED collection. Results show that MUDOF2 can not only improve the classification performances of the component algorithms, but also largely improve the Linear Combination under the Weighting Strategy Based On Utility Measure.

9.2 Future Work

The performance of our proposed MUDOF approaches for automatic text categorization has already been shown to demonstrate improvement of classification performances. More research can be done to further explore its potential classification efficiency. The direction of further studies include the followings:

- More advanced feature characteristics can be collected. Although, performance of MUDOF and MUDOF2 are already significant by using our proposed feature characteristics, the predictive accuracy can be increased further with more representative and domain specific document feature characteristics.

- Different regression models can be employed. In addition to the linear regression models, other types of regression models can be tried for finding the relationship between document feature characteristics and the classification errors.

Appendix A

Details of Experimental Results for Reuters-21578 corpus

Category	RO	WH	KNN	SVM	GISR	GISW	LC1
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.947
alum	0.766	0.766	0.681	0.766	0.765	0.751	0.751
barley	0.483	0.760	0.552	0.690	0.670	0.898	0.651
bop	0.525	0.646	0.590	0.689	0.646	0.623	0.656
carcass	0.703	0.739	0.649	0.649	0.649	0.686	0.703
castor-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cocoa	0.920	0.974	0.920	0.974	0.920	0.974	0.950
coconut	0.833	0.833	0.833	0.833	0.833	0.833	0.833
coconut-oil	0.292	0.583	0.583	0.292	0.583	0.583	0.583
coffee	0.913	0.913	0.898	0.913	0.877	0.913	0.913
copper	0.865	0.865	0.865	0.844	0.865	0.844	0.865
copra-cake	0.500	0.500	0.500	0.500	0.500	0.500	0.500
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.846
cotton	0.732	0.781	0.635	0.683	0.764	0.830	0.732
cotton-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cpi	0.386	0.562	0.421	0.621	0.449	0.597	0.491
cpu	0.750	0.750	0.750	0.750	0.750	0.750	0.750
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.860
dfl	0.500	0.500	0.500	0.500	0.500	0.500	0.500
dlr	0.652	0.748	0.645	0.697	0.734	0.742	0.742
dmk	0.225	0.225	0.225	0.000	0.208	0.225	0.208
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.979
fuel	0.286	0.286	0.367	0.477	0.382	0.550	0.477
To be cont'd ...							

Table cont'd ...							
gas	0.446	0.490	0.446	0.743	0.515	0.743	0.724
gnp	0.723	0.845	0.845	0.873	0.890	0.845	0.873
gold	0.853	0.820	0.787	0.820	0.820	0.820	0.853
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.896
groundnut	0.000	0.000	0.000	0.250	0.000	0.250	0.000
groundnut-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
heat	0.550	0.550	0.550	0.550	0.550	0.550	0.550
hog	0.774	0.774	0.774	0.774	0.774	0.774	0.774
housing	0.675	0.675	0.675	0.675	0.675	0.675	0.675
income	0.536	0.804	0.670	0.670	0.804	0.670	0.670
instal-debt	0.750	0.750	0.750	0.750	0.750	0.750	0.750
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.790
ipi	0.481	0.881	0.721	0.962	0.721	0.881	0.881
iron-steel	0.690	0.690	0.737	0.760	0.690	0.690	0.690
jet	0.000	0.000	0.000	0.600	0.000	0.000	0.000
jobs	0.605	0.931	0.732	0.745	0.884	0.931	0.884
l-cattle	0.350	0.350	0.321	0.417	0.350	0.417	0.375
lead	0.670	0.670	0.670	0.402	0.552	0.552	0.621
lei	0.875	0.875	0.875	0.875	0.875	0.875	0.875
lin-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
livestock	0.721	0.735	0.694	0.694	0.801	0.735	0.694
lumber	0.464	0.464	0.464	0.774	0.464	0.774	0.619
meal-feed	0.359	0.770	0.564	0.616	0.667	0.718	0.667
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.763
money-supply	0.515	0.696	0.743	0.754	0.667	0.743	0.772
naphtha	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nat-gas	0.525	0.656	0.623	0.656	0.689	0.689	0.656
nickel	0.750	0.750	0.750	0.750	0.750	0.750	0.750
nkr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nzdlr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
oat	0.310	0.464	0.310	0.310	0.310	0.464	0.310
oilseed	0.511	0.716	0.604	0.702	0.674	0.737	0.722
orange	0.839	0.871	0.784	0.784	0.871	0.871	0.871
palladium	0.500	0.500	0.500	0.500	0.500	0.500	0.500
palm-oil	0.764	0.764	0.764	0.764	0.668	0.764	0.764
palmkernel	0.500	0.500	0.500	0.500	0.500	0.500	0.500
pet-chem	0.401	0.321	0.401	0.321	0.401	0.401	0.401
platinum	0.786	0.786	0.786	0.786	0.786	0.786	0.786
potato	0.875	0.875	0.875	0.875	0.875	0.875	0.800
To be cont'd ...							

Table cont'd ...							
propane	0.583	0.875	0.583	0.875	0.583	0.583	0.583
rand	0.500	0.500	0.500	0.500	0.500	0.500	0.500
rape-oil	0.000	0.000	0.292	0.292	0.000	0.000	0.292
rapeseed	0.505	0.739	0.528	0.633	0.633	0.844	0.633
reserves	0.703	0.703	0.703	0.811	0.757	0.703	0.757
retail	0.375	0.417	0.417	0.417	0.417	0.417	0.417
rice	0.613	0.826	0.694	0.776	0.735	0.776	0.801
rubber	0.721	0.851	0.721	0.881	0.721	0.801	0.721
rye	0.500	0.500	0.500	0.500	0.500	0.500	0.500
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.884
silver	0.590	0.826	0.590	0.708	0.590	0.708	0.708
sorghum	0.477	0.573	0.382	0.550	0.573	0.573	0.550
soy-meal	0.519	0.816	0.668	0.593	0.574	0.593	0.742
soy-oil	0.261	0.261	0.252	0.174	0.348	0.174	0.261
soybean	0.537	0.746	0.602	0.627	0.736	0.746	0.717
strategic-metal	0.087	0.087	0.087	0.174	0.087	0.087	0.087
sugar	0.685	0.877	0.795	0.784	0.767	0.822	0.795
sun-meal	0.500	0.500	0.500	0.500	0.500	0.500	0.500
sun-oil	0.417	0.417	0.417	0.417	0.417	0.417	0.417
sunseed	0.367	0.183	0.367	0.171	0.367	0.367	0.367
tea	0.625	0.675	0.450	0.675	0.675	0.675	0.675
tin	0.881	0.962	0.881	0.881	0.881	0.962	0.851
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.792
veg-oil	0.613	0.720	0.587	0.613	0.553	0.667	0.747
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.825
wpi	0.573	0.668	0.764	0.668	0.764	0.668	0.764
yen	0.276	0.483	0.536	0.276	0.469	0.402	0.483
zinc	0.964	0.890	0.964	0.862	0.890	0.890	0.964
All MBE	0.776	0.820	0.802	0.841	0.830	0.845	0.860
All ABE	0.578	0.649	0.607	0.640	0.625	0.655	0.647
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.858
Other ABE	0.556	0.625	0.585	0.617	0.601	0.630	0.621

Table A.1: Complete comparison of Linear Combination approach under the Equal Weighting Strategy (LC1) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	LC2
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.947
alum	0.766	0.766	0.681	0.766	0.765	0.751	0.751
barley	0.483	0.760	0.552	0.690	0.670	0.898	0.760
bop	0.525	0.646	0.590	0.689	0.646	0.623	0.646
carcass	0.703	0.739	0.649	0.649	0.649	0.686	0.703
castor-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cocoa	0.920	0.974	0.920	0.974	0.920	0.974	0.974
coconut	0.833	0.833	0.833	0.833	0.833	0.833	0.833
coconut-oil	0.292	0.583	0.583	0.292	0.583	0.583	0.500
coffee	0.913	0.913	0.898	0.913	0.877	0.913	0.913
copper	0.865	0.865	0.865	0.844	0.865	0.844	0.865
copra-cake	0.500	0.500	0.500	0.500	0.500	0.500	0.500
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.853
cotton	0.732	0.781	0.635	0.683	0.764	0.830	0.781
cotton-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cpi	0.386	0.562	0.421	0.621	0.449	0.597	0.518
cpu	0.750	0.750	0.750	0.750	0.750	0.750	0.750
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.860
dfl	0.500	0.500	0.500	0.500	0.500	0.500	0.500
dlr	0.652	0.748	0.645	0.697	0.734	0.742	0.726
dmk	0.225	0.225	0.225	0.000	0.208	0.225	0.208
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.979
fuel	0.286	0.286	0.367	0.477	0.382	0.550	0.442
gas	0.446	0.490	0.446	0.743	0.515	0.743	0.629
gnp	0.723	0.845	0.845	0.873	0.890	0.845	0.873
gold	0.853	0.820	0.787	0.820	0.820	0.820	0.853
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.897
groundnut	0.000	0.000	0.000	0.250	0.000	0.250	0.000
groundnut-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
heat	0.550	0.550	0.550	0.550	0.550	0.550	0.550
hog	0.774	0.774	0.774	0.774	0.774	0.774	0.774
housing	0.675	0.675	0.675	0.675	0.675	0.675	0.675
income	0.536	0.804	0.670	0.670	0.804	0.670	0.670
instal-debt	0.750	0.750	0.750	0.750	0.750	0.750	0.750
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.796
ipi	0.481	0.881	0.721	0.962	0.721	0.881	0.881
iron-steel	0.690	0.690	0.737	0.760	0.690	0.690	0.690
jet	0.000	0.000	0.000	0.600	0.000	0.000	0.000
To be cont'd ...							

Table cont'd ...							
jobs	0.605	0.931	0.732	0.745	0.884	0.931	0.884
l-cattle	0.350	0.350	0.321	0.417	0.350	0.417	0.375
lead	0.670	0.670	0.670	0.402	0.552	0.552	0.621
lei	0.875	0.875	0.875	0.875	0.875	0.875	0.875
lin-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
livestock	0.721	0.735	0.694	0.694	0.801	0.735	0.721
lumber	0.464	0.464	0.464	0.774	0.464	0.774	0.729
meal-feed	0.359	0.770	0.564	0.616	0.667	0.718	0.718
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.764
money-supply	0.515	0.696	0.743	0.754	0.667	0.743	0.783
naphtha	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nat-gas	0.525	0.656	0.623	0.656	0.689	0.689	0.656
nickel	0.750	0.750	0.750	0.750	0.750	0.750	0.750
nkr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nzdlr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
oat	0.310	0.464	0.310	0.310	0.310	0.464	0.310
oilseed	0.511	0.716	0.604	0.702	0.674	0.737	0.729
orange	0.839	0.871	0.784	0.784	0.871	0.871	0.871
palladium	0.500	0.500	0.500	0.500	0.500	0.500	0.500
palm-oil	0.764	0.764	0.764	0.764	0.668	0.764	0.764
palmkernel	0.500	0.500	0.500	0.500	0.500	0.500	0.500
pet-chem	0.401	0.321	0.401	0.321	0.401	0.401	0.401
platinum	0.786	0.786	0.786	0.786	0.786	0.786	0.786
potato	0.875	0.875	0.875	0.875	0.875	0.875	0.500
propane	0.583	0.875	0.583	0.875	0.583	0.583	0.583
rand	0.500	0.500	0.500	0.500	0.500	0.500	0.500
rape-oil	0.000	0.000	0.292	0.292	0.000	0.000	0.292
rapeseed	0.505	0.739	0.528	0.633	0.633	0.844	0.633
reserves	0.703	0.703	0.703	0.811	0.757	0.703	0.757
retail	0.375	0.417	0.417	0.417	0.417	0.417	0.417
rice	0.613	0.826	0.694	0.776	0.735	0.776	0.801
rubber	0.721	0.851	0.721	0.881	0.721	0.801	0.721
rye	0.500	0.500	0.500	0.500	0.500	0.500	0.500
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.883
silver	0.590	0.826	0.590	0.708	0.590	0.708	0.826
sorghum	0.477	0.573	0.382	0.550	0.573	0.573	0.573
soy-meal	0.519	0.816	0.668	0.593	0.574	0.593	0.742
soy-oil	0.261	0.261	0.252	0.174	0.348	0.174	0.336
soybean	0.537	0.746	0.602	0.627	0.736	0.746	0.717
To be cont'd ...							

Table cont'd ...							
strategic-metal	0.087	0.087	0.087	0.174	0.087	0.087	0.168
sugar	0.685	0.877	0.795	0.784	0.767	0.822	0.795
sun-meal	0.500	0.500	0.500	0.500	0.500	0.500	0.500
sun-oil	0.417	0.417	0.417	0.417	0.417	0.417	0.375
sunseed	0.367	0.183	0.367	0.171	0.367	0.367	0.343
tea	0.625	0.675	0.450	0.675	0.675	0.675	0.675
tin	0.881	0.962	0.881	0.881	0.881	0.962	0.881
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.797
veg-oil	0.613	0.720	0.587	0.613	0.553	0.667	0.728
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.839
wpi	0.573	0.668	0.764	0.668	0.764	0.668	0.764
yen	0.276	0.483	0.536	0.276	0.469	0.402	0.483
zinc	0.964	0.890	0.964	0.862	0.890	0.890	0.964
All MBE	0.776	0.820	0.802	0.841	0.830	0.845	0.861
All ABE	0.578	0.649	0.607	0.640	0.625	0.655	0.649
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.862
Other ABE	0.556	0.625	0.585	0.617	0.601	0.630	0.622

Table A.2: Complete comparison of Linear Combination approach under the Weighting Strategy Based On Utility Measure (LC2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	LC3
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.949
alum	0.766	0.766	0.681	0.766	0.765	0.751	0.751
barley	0.483	0.760	0.552	0.690	0.670	0.898	0.690
bop	0.525	0.646	0.590	0.689	0.646	0.623	0.656
carcass	0.703	0.739	0.649	0.649	0.649	0.686	0.703
castor-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cocoa	0.920	0.974	0.920	0.974	0.920	0.974	0.920
coconut	0.833	0.833	0.833	0.833	0.833	0.833	0.833
coconut-oil	0.292	0.583	0.583	0.292	0.583	0.583	0.583
coffee	0.913	0.913	0.898	0.913	0.877	0.913	0.913
copper	0.865	0.865	0.865	0.844	0.865	0.844	0.865
copra-cake	0.500	0.500	0.500	0.500	0.500	0.500	0.500
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.867
cotton	0.732	0.781	0.635	0.683	0.764	0.830	0.716
cotton-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cpi	0.386	0.562	0.421	0.621	0.449	0.597	0.491
cpu	0.750	0.750	0.750	0.750	0.750	0.750	0.750
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.860
dfl	0.500	0.500	0.500	0.500	0.500	0.500	0.500
dlr	0.652	0.748	0.645	0.697	0.734	0.742	0.719
dmk	0.225	0.225	0.225	0.000	0.208	0.225	0.225
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.978
fuel	0.286	0.286	0.367	0.477	0.382	0.550	0.477
gas	0.446	0.490	0.446	0.743	0.515	0.743	0.745
gnp	0.723	0.845	0.845	0.873	0.890	0.845	0.873
gold	0.853	0.820	0.787	0.820	0.820	0.820	0.853
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.896
groundnut	0.000	0.000	0.000	0.250	0.000	0.250	0.000
groundnut-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
heat	0.550	0.550	0.550	0.550	0.550	0.550	0.550
hog	0.774	0.774	0.774	0.774	0.774	0.774	0.729
housing	0.675	0.675	0.675	0.675	0.675	0.675	0.675
income	0.536	0.804	0.670	0.670	0.804	0.670	0.670
instal-debt	0.750	0.750	0.750	0.750	0.750	0.750	0.750
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.794
ipi	0.481	0.881	0.721	0.962	0.721	0.881	0.851
iron-steel	0.690	0.690	0.737	0.760	0.690	0.690	0.690
jet	0.000	0.000	0.000	0.600	0.000	0.000	0.000
To be cont'd ...							

Table cont'd ...							
jobs	0.605	0.931	0.732	0.745	0.884	0.931	0.838
l-cattle	0.350	0.350	0.321	0.417	0.350	0.417	0.350
lead	0.670	0.670	0.670	0.402	0.552	0.552	0.621
lei	0.875	0.875	0.875	0.875	0.875	0.875	0.875
lin-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
livestock	0.721	0.735	0.694	0.694	0.801	0.735	0.641
lumber	0.464	0.464	0.464	0.774	0.464	0.774	0.619
meal-feed	0.359	0.770	0.564	0.616	0.667	0.718	0.652
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.758
money-supply	0.515	0.696	0.743	0.754	0.667	0.743	0.783
naphtha	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nat-gas	0.525	0.656	0.623	0.656	0.689	0.689	0.656
nickel	0.750	0.750	0.750	0.750	0.750	0.750	0.750
nkr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nzdlr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
oat	0.310	0.464	0.310	0.310	0.310	0.464	0.310
oilseed	0.511	0.716	0.604	0.702	0.674	0.737	0.722
orange	0.839	0.871	0.784	0.784	0.871	0.871	0.871
palladium	0.500	0.500	0.500	0.500	0.500	0.500	0.500
palm-oil	0.764	0.764	0.764	0.764	0.668	0.764	0.764
palmkernel	0.500	0.500	0.500	0.500	0.500	0.500	0.500
pet-chem	0.401	0.321	0.401	0.321	0.401	0.401	0.401
platinum	0.786	0.786	0.786	0.786	0.786	0.786	0.786
potato	0.875	0.875	0.875	0.875	0.875	0.875	0.875
propane	0.583	0.875	0.583	0.875	0.583	0.583	0.583
rand	0.500	0.500	0.500	0.500	0.500	0.500	0.500
rape-oil	0.000	0.000	0.292	0.292	0.000	0.000	0.292
rapeseed	0.505	0.739	0.528	0.633	0.633	0.844	0.633
reserves	0.703	0.703	0.703	0.811	0.757	0.703	0.757
retail	0.375	0.417	0.417	0.417	0.417	0.417	0.417
rice	0.613	0.826	0.694	0.776	0.735	0.776	0.776
rubber	0.721	0.851	0.721	0.881	0.721	0.801	0.721
rye	0.500	0.500	0.500	0.500	0.500	0.500	0.500
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.883
silver	0.590	0.826	0.590	0.708	0.590	0.708	0.708
sorghum	0.477	0.573	0.382	0.550	0.573	0.573	0.531
soy-meal	0.519	0.816	0.668	0.593	0.574	0.593	0.668
soy-oil	0.261	0.261	0.252	0.174	0.348	0.174	0.261
soybean	0.537	0.746	0.602	0.627	0.736	0.746	0.717
To be cont'd ...							

Table cont'd ...							
strategic-metal	0.087	0.087	0.087	0.174	0.087	0.087	0.087
sugar	0.685	0.877	0.795	0.784	0.767	0.822	0.795
sun-meal	0.500	0.500	0.500	0.500	0.500	0.500	0.500
sun-oil	0.417	0.417	0.417	0.417	0.417	0.417	0.417
sunseed	0.367	0.183	0.367	0.171	0.367	0.367	0.367
tea	0.625	0.675	0.450	0.675	0.675	0.675	0.675
tin	0.881	0.962	0.881	0.881	0.881	0.962	0.881
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.784
veg-oil	0.613	0.720	0.587	0.613	0.553	0.667	0.702
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.825
wpi	0.573	0.668	0.764	0.668	0.764	0.668	0.733
yen	0.276	0.483	0.536	0.276	0.469	0.402	0.483
zinc	0.964	0.890	0.964	0.862	0.890	0.890	0.964
All MBE	0.776	0.820	0.802	0.841	0.830	0.845	0.858
All ABE	0.578	0.649	0.607	0.640	0.625	0.655	0.644
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.859
Other ABE	0.556	0.625	0.585	0.617	0.601	0.630	0.617

Table A.3: Complete comparison of Linear Combination approach under the Weighting Strategy Based On Document Rank (LC3) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	MUDOF	IDEAL
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.909	0.932
alum	0.766	0.766	0.681	0.766	0.765	0.751	0.766	0.766
barley	0.483	0.760	0.552	0.690	0.670	0.898	0.898	0.898
bop	0.525	0.646	0.590	0.689	0.646	0.623	0.623	0.689
carcass	0.703	0.739	0.649	0.649	0.649	0.686	0.739	0.739
castor-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cocoa	0.920	0.974	0.920	0.974	0.920	0.974	0.974	0.974
coconut	0.833	0.833	0.833	0.833	0.833	0.833	0.833	0.833
coconut-oil	0.292	0.583	0.583	0.292	0.583	0.583	0.583	0.583
coffee	0.913	0.913	0.898	0.913	0.877	0.913	0.913	0.913
copper	0.865	0.865	0.865	0.844	0.865	0.844	0.844	0.865
copra-cake	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.885	0.885
cotton	0.732	0.781	0.635	0.683	0.764	0.830	0.830	0.830
cotton-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cpi	0.386	0.562	0.421	0.621	0.449	0.597	0.597	0.621
cpu	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.853	0.871
dfl	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
dlr	0.652	0.748	0.645	0.697	0.734	0.742	0.742	0.748
dmk	0.225	0.225	0.225	0.000	0.208	0.225	0.225	0.225
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.980	0.980
fuel	0.286	0.286	0.367	0.477	0.382	0.550	0.477	0.550
gas	0.446	0.490	0.446	0.743	0.515	0.743	0.743	0.743
gnp	0.723	0.845	0.845	0.873	0.890	0.845	0.845	0.890
gold	0.853	0.820	0.787	0.820	0.820	0.820	0.820	0.853
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.910	0.917
groundnut	0.000	0.000	0.000	0.250	0.000	0.250	0.000	0.250
groundnut-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
heat	0.550	0.550	0.550	0.550	0.550	0.550	0.550	0.550
hog	0.774	0.774	0.774	0.774	0.774	0.774	0.774	0.774
housing	0.675	0.675	0.675	0.675	0.675	0.675	0.675	0.675
income	0.536	0.804	0.670	0.670	0.804	0.670	0.670	0.804
instal-debt	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.745	0.758
ipi	0.481	0.881	0.721	0.962	0.721	0.881	0.881	0.962
iron-steel	0.690	0.690	0.737	0.760	0.690	0.690	0.690	0.760
jet	0.000	0.000	0.000	0.600	0.000	0.000	0.600	0.600
To be cont'd ...								

Table cont'd ...							
jobs	0.605	0.931	0.732	0.745	0.884	0.931	0.931	0.931
l-cattle	0.350	0.350	0.321	0.417	0.350	0.417	0.350	0.417
lead	0.670	0.670	0.670	0.402	0.552	0.552	0.402	0.670
lei	0.875	0.875	0.875	0.875	0.875	0.875	0.875	0.875
lin-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
livestock	0.721	0.735	0.694	0.694	0.801	0.735	0.735	0.801
lumber	0.464	0.464	0.464	0.774	0.464	0.774	0.774	0.774
meal-feed	0.359	0.770	0.564	0.616	0.667	0.718	0.770	0.770
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.756	0.756
money-supply	0.515	0.696	0.743	0.754	0.667	0.743	0.743	0.754
naphtha	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nat-gas	0.525	0.656	0.623	0.656	0.689	0.689	0.689	0.689
nickel	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
nkr	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nzdlr	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
oat	0.310	0.464	0.310	0.310	0.310	0.464	0.464	0.464
oilseed	0.511	0.716	0.604	0.702	0.674	0.737	0.737	0.716
orange	0.839	0.871	0.784	0.784	0.871	0.871	0.871	0.871
palladium	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
palm-oil	0.764	0.764	0.764	0.764	0.668	0.764	0.764	0.764
palmkernel	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
pet-chem	0.401	0.321	0.401	0.321	0.401	0.401	0.321	0.401
platinum	0.786	0.786	0.786	0.786	0.786	0.786	0.786	0.786
potato	0.875	0.875	0.875	0.875	0.875	0.875	0.875	0.875
propane	0.583	0.875	0.583	0.875	0.583	0.583	0.875	0.875
rand	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
rape-oil	0.000	0.000	0.292	0.292	0.000	0.000	0.000	0.292
rapeseed	0.505	0.739	0.528	0.633	0.633	0.844	0.739	0.844
reserves	0.703	0.703	0.703	0.811	0.757	0.703	0.703	0.811
retail	0.375	0.417	0.417	0.417	0.417	0.417	0.417	0.417
rice	0.613	0.826	0.694	0.776	0.735	0.776	0.826	0.826
rubber	0.721	0.851	0.721	0.881	0.721	0.801	0.881	0.881
rye	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.872	0.872
silver	0.590	0.826	0.590	0.708	0.590	0.708	0.708	0.826
sorghum	0.477	0.573	0.382	0.550	0.573	0.573	0.573	0.573
soy-meal	0.519	0.816	0.668	0.593	0.574	0.593	0.593	0.816
soy-oil	0.261	0.261	0.252	0.174	0.348	0.174	0.261	0.348
soybean	0.537	0.746	0.602	0.627	0.736	0.746	0.746	0.746
To be cont'd ...								

Table cont'd ...								
strategic-metal	0.087	0.087	0.087	0.174	0.087	0.087	0.087	0.174
sugar	0.685	0.877	0.795	0.784	0.767	0.822	0.822	0.877
sun-meal	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
sun-oil	0.417	0.417	0.417	0.417	0.417	0.417	0.417	0.417
sunseed	0.367	0.183	0.367	0.171	0.367	0.367	0.183	0.367
tea	0.625	0.675	0.450	0.675	0.675	0.675	0.675	0.675
tin	0.881	0.962	0.881	0.881	0.881	0.962	0.962	0.962
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.788	0.788
veg-oil	0.613	0.720	0.587	0.613	0.553	0.667	0.667	0.720
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.875	0.875
wpi	0.573	0.668	0.764	0.668	0.764	0.668	0.668	0.764
yen	0.276	0.483	0.536	0.276	0.469	0.402	0.402	0.536
zinc	0.964	0.890	0.964	0.862	0.890	0.890	0.890	0.964
All MBE	0.776	0.820	0.802	0.841	0.830	0.845	0.847	0.868
All ABE	0.578	0.649	0.607	0.640	0.625	0.655	0.659	0.692
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.857	0.863
Other ABE	0.556	0.625	0.585	0.617	0.601	0.630	0.634	0.670

Table A.4: Complete comparison of MUDOF approach (MUDOF) and the ideal combination of algorithms (IDEAL) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	MUDOF2
acq	0.829	0.870	0.859	0.931	0.932	0.909	0.947
alum	0.766	0.766	0.681	0.766	0.765	0.751	0.751
barley	0.483	0.760	0.552	0.690	0.670	0.898	0.737
bop	0.525	0.646	0.590	0.689	0.646	0.623	0.689
carcass	0.703	0.739	0.649	0.649	0.649	0.686	0.703
castor-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cocoa	0.920	0.974	0.920	0.974	0.920	0.974	0.974
coconut	0.833	0.833	0.833	0.833	0.833	0.833	0.833
coconut-oil	0.292	0.583	0.583	0.292	0.583	0.583	0.583
coffee	0.913	0.913	0.898	0.913	0.877	0.913	0.913
copper	0.865	0.865	0.865	0.844	0.865	0.844	0.865
copra-cake	0.500	0.500	0.500	0.500	0.500	0.500	0.500
corn	0.614	0.867	0.690	0.832	0.867	0.885	0.867
cotton	0.732	0.781	0.635	0.683	0.764	0.830	0.764
cotton-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
cpi	0.386	0.562	0.421	0.621	0.449	0.597	0.552
cpu	0.750	0.750	0.750	0.750	0.750	0.750	0.667
crude	0.793	0.853	0.823	0.871	0.813	0.869	0.863
dfl	0.500	0.500	0.500	0.500	0.500	0.500	0.500
dlr	0.652	0.748	0.645	0.697	0.734	0.742	0.734
dmk	0.225	0.225	0.225	0.000	0.208	0.225	0.225
earn	0.956	0.969	0.956	0.980	0.959	0.962	0.979
fuel	0.286	0.286	0.367	0.477	0.382	0.550	0.442
gas	0.446	0.490	0.446	0.743	0.515	0.743	0.858
gnp	0.723	0.845	0.845	0.873	0.890	0.845	0.873
gold	0.853	0.820	0.787	0.820	0.820	0.820	0.853
grain	0.803	0.887	0.820	0.917	0.804	0.910	0.908
groundnut	0.000	0.000	0.000	0.250	0.000	0.250	0.000
groundnut-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
heat	0.550	0.550	0.550	0.550	0.550	0.550	0.550
hog	0.774	0.774	0.774	0.774	0.774	0.774	0.774
housing	0.675	0.675	0.675	0.675	0.675	0.675	0.675
income	0.536	0.804	0.670	0.670	0.804	0.670	0.670
instal-debt	0.750	0.750	0.750	0.750	0.750	0.750	0.750
interest	0.702	0.749	0.712	0.619	0.758	0.745	0.785
ipi	0.481	0.881	0.721	0.962	0.721	0.881	0.881
iron-steel	0.690	0.690	0.737	0.760	0.690	0.690	0.690
jet	0.000	0.000	0.000	0.600	0.000	0.000	0.000
To be cont'd ...							

Table cont'd ...							
jobs	0.605	0.931	0.732	0.745	0.884	0.931	0.865
l-cattle	0.350	0.350	0.321	0.417	0.350	0.417	0.417
lead	0.670	0.670	0.670	0.402	0.552	0.552	0.621
lei	0.875	0.875	0.875	0.875	0.875	0.875	0.875
lin-oil	0.500	0.500	0.500	0.500	0.500	0.500	0.500
livestock	0.721	0.735	0.694	0.694	0.801	0.735	0.694
lumber	0.464	0.464	0.464	0.774	0.464	0.774	0.619
meal-feed	0.359	0.770	0.564	0.616	0.667	0.718	0.718
money-fx	0.582	0.718	0.674	0.717	0.681	0.756	0.756
money-supply	0.515	0.696	0.743	0.754	0.667	0.743	0.783
naphtha	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nat-gas	0.525	0.656	0.623	0.656	0.689	0.689	0.689
nickel	0.750	0.750	0.750	0.750	0.750	0.750	0.750
nkr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
nzdlr	0.500	0.500	0.500	0.500	0.500	0.500	0.500
oat	0.310	0.464	0.310	0.310	0.310	0.464	0.310
oilseed	0.511	0.716	0.604	0.702	0.674	0.737	0.737
orange	0.839	0.871	0.784	0.784	0.871	0.871	0.839
palladium	0.500	0.500	0.500	0.500	0.500	0.500	0.500
palm-oil	0.764	0.764	0.764	0.764	0.668	0.764	0.764
palmkernel	0.500	0.500	0.500	0.500	0.500	0.500	0.500
pet-chem	0.401	0.321	0.401	0.321	0.401	0.401	0.387
platinum	0.786	0.786	0.786	0.786	0.786	0.786	0.786
potato	0.875	0.875	0.875	0.875	0.875	0.875	0.875
propane	0.583	0.875	0.583	0.875	0.583	0.583	0.583
rand	0.500	0.500	0.500	0.500	0.500	0.500	0.500
rape-oil	0.000	0.000	0.292	0.292	0.000	0.000	0.292
rapeseed	0.505	0.739	0.528	0.633	0.633	0.844	0.633
reserves	0.703	0.703	0.703	0.811	0.757	0.703	0.757
retail	0.375	0.417	0.417	0.417	0.417	0.417	0.417
rice	0.613	0.826	0.694	0.776	0.735	0.776	0.817
rubber	0.721	0.851	0.721	0.881	0.721	0.801	0.721
rye	0.500	0.500	0.500	0.500	0.500	0.500	0.500
ship	0.800	0.860	0.800	0.845	0.825	0.872	0.878
silver	0.590	0.826	0.590	0.708	0.590	0.708	0.826
sorghum	0.477	0.573	0.382	0.550	0.573	0.573	0.573
soy-meal	0.519	0.816	0.668	0.593	0.574	0.593	0.668
soy-oil	0.261	0.261	0.252	0.174	0.348	0.174	0.261
soybean	0.537	0.746	0.602	0.627	0.736	0.746	0.717
To be cont'd ...							

Table cont'd ...							
strategic-metal	0.087	0.087	0.087	0.174	0.087	0.087	0.087
sugar	0.685	0.877	0.795	0.784	0.767	0.822	0.795
sun-meal	0.500	0.500	0.500	0.500	0.500	0.500	0.500
sun-oil	0.417	0.417	0.417	0.417	0.417	0.417	0.417
sunseed	0.367	0.183	0.367	0.171	0.367	0.367	0.367
tea	0.625	0.675	0.450	0.675	0.675	0.675	0.675
tin	0.881	0.962	0.881	0.881	0.881	0.962	0.881
trade	0.732	0.763	0.740	0.715	0.714	0.788	0.778
veg-oil	0.613	0.720	0.587	0.613	0.553	0.667	0.720
wheat	0.713	0.839	0.727	0.820	0.825	0.875	0.847
wpi	0.573	0.668	0.764	0.668	0.764	0.668	0.764
yen	0.276	0.483	0.536	0.276	0.469	0.402	0.469
zinc	0.964	0.890	0.964	0.862	0.890	0.890	0.964
All MBE	0.776	0.820	0.802	0.841	0.830	0.845	0.863
All ABE	0.578	0.649	0.607	0.640	0.625	0.655	0.653
Top 10 ABE	0.752	0.838	0.780	0.825	0.818	0.857	0.861
Other ABE	0.556	0.625	0.585	0.617	0.601	0.630	0.627

Table A.5: Complete comparison of MUDOF2 approach (MUDOF2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the Reuters-21578 corpus.

Appendix B

Details of Experimental Results for OHSUMED corpus

Category	RO	WH	KNN	SVM	GISR	GISW	LC1
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.485
Angina Pectoris, Variant	0.238	0.583	0.238	0.222	0.292	0.229	0.229
Angina, Unstable	0.725	0.870	0.772	0.783	0.870	0.783	0.870
Aortic Coarctation	0.816	0.964	0.742	0.742	0.890	0.862	0.816
Aortic Subvalvular Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Aortic Valve Insufficiency	0.528	0.633	0.633	0.633	0.739	0.739	0.633
Aortic Valve Stenosis	0.517	0.387	0.517	0.565	0.452	0.387	0.502
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.580
Atrial Fibrillation	0.452	0.710	0.387	0.646	0.581	0.646	0.581
Atrial Flutter	0.641	0.881	0.641	0.851	0.774	0.881	0.801
Bradycardia	0.477	0.573	0.477	0.382	0.573	0.668	0.668
Bundle-Branch Block	0.826	0.708	0.826	0.826	0.708	0.788	0.788
Carcinoid Heart Disease	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cardiac Output, Low	0.076	0.069	0.091	0.071	0.074	0.000	0.081
Cardiac Tamponade	0.573	0.668	0.477	0.668	0.477	0.668	0.668
Cardiomyopathy, Congestive	0.372	0.558	0.491	0.512	0.465	0.512	0.547
Cardiomyopathy, Hypertrophic	0.422	0.528	0.317	0.211	0.528	0.486	0.422
Cardiomyopathy, Restrictive	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Chagas Cardiomyopathy	0.450	0.750	0.250	0.750	0.750	0.750	0.750
Cor Triatriatum	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Coronary Aneurysm	0.464	0.155	0.619	0.619	0.619	0.155	0.464
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.400
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.579
Coronary Thrombosis	0.377	0.445	0.415	0.453	0.377	0.415	0.445
Coronary Vasospasm	0.171	0.367	0.183	0.367	0.550	0.514	0.550
Coronary Vessel Anomalies	0.464	0.583	0.774	0.774	0.929	0.619	0.619
Double Outlet Right Ventricle	0.000	0.000	0.625	0.000	0.000	0.000	0.000
Ductus Arteriosus, Patent	0.875	0.875	0.875	0.583	0.875	0.875	0.875
To be cont'd ...							

Table cont'd ...							
Ebstein's Anomaly	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Eisenmenger Complex	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Endocarditis	0.183	0.367	0.183	0.183	0.171	0.367	0.183
Endocarditis, Bacterial	0.621	0.552	0.552	0.737	0.621	0.760	0.621
Endomyocardial Fibrosis	0.292	0.000	0.000	0.292	0.000	0.292	0.292
Extrasystole	0.268	0.268	0.402	0.134	0.381	0.134	0.268
Heart Aneurysm	0.310	0.155	0.464	0.438	0.292	0.155	0.464
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.638
Heart Block	0.422	0.292	0.422	0.528	0.422	0.528	0.528
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.667
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.310
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.603
Heart Murmurs	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Neoplasms	0.536	0.402	0.536	0.268	0.402	0.381	0.536
Heart Rupture	0.550	0.367	0.367	0.550	0.550	0.367	0.550
Heart Rupture, Post-Infarction	0.500	0.250	0.500	0.500	0.500	0.250	0.500
Heart Septal Defects	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Septal Defects, Atrial	0.583	0.583	0.583	0.583	0.800	0.583	0.583
Heart Septal Defects, Ventricular	0.438	0.464	0.464	0.438	0.464	0.464	0.464
Heart Valve Diseases	0.207	0.483	0.276	0.414	0.469	0.483	0.483
Kearns Syndrome	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Long QT Syndrome	0.583	0.583	0.875	0.583	0.875	0.583	0.583
Mitral Valve Insufficiency	0.414	0.483	0.402	0.621	0.483	0.621	0.621
Mitral Valve Prolapse	0.292	0.583	0.583	0.583	0.583	0.583	0.583
Mitral Valve Stenosis	0.515	0.707	0.572	0.686	0.629	0.743	0.686
Myocardial Diseases	0.345	0.402	0.276	0.276	0.276	0.207	0.414
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.812
Myocarditis	0.155	0.310	0.464	0.464	0.464	0.464	0.464
Pericardial Effusion	0.550	0.550	0.550	0.367	0.550	0.550	0.550
Pericarditis	0.183	0.183	0.183	0.550	0.550	0.367	0.550
Pericarditis, Constrictive	0.550	0.733	0.550	0.514	0.550	0.367	0.550
Pulmonary Heart Disease	0.583	0.875	0.583	0.583	0.583	0.583	0.583
Pulmonary Valve Insufficiency	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Pulmonary Valve Stenosis	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Rheumatic Heart Disease	0.250	0.267	0.267	0.292	0.292	0.000	0.583
Shock, Cardiogenic	0.590	0.590	0.590	0.590	0.472	0.590	0.590
Sick Sinus Syndrome	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.684
Tachycardia, Atrioventricular Nodal Reentry	0.225	0.196	0.225	0.225	0.225	0.225	0.225
Tachycardia, Ectopic Atrial	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Ectopic Junctional	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Paroxysmal	0.619	0.464	0.464	0.619	0.619	0.619	0.619
Tachycardia, Supraventricular	0.586	0.668	0.537	0.716	0.683	0.716	0.683
Tetralogy of Fallot	0.583	0.583	0.583	0.533	0.583	0.583	0.583
Transposition of Great Vessels	0.225	0.417	0.450	0.450	0.450	0.225	0.450
Tricuspid Valve Insufficiency	0.183	0.343	0.171	0.367	0.183	0.367	0.183
Tricuspid Valve Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
To be cont'd ...							

Table cont'd ...							
Truncus Arteriosus, Persistent	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Ventricular Fibrillation	0.297	0.445	0.371	0.431	0.297	0.431	0.445
Ventricular Outflow Obstruction	0.417	0.417	0.417	0.833	0.417	0.417	0.417
Wolff-Parkinson-White Syndrome	0.675	0.625	0.675	0.675	0.675	0.675	0.675
Myocardial Reperfusion Injury	0.489	0.489	0.534	0.400	0.489	0.578	0.523
Torsades de Pointes	0.750	0.750	0.750	0.750	0.750	0.750	0.750
All MBE	0.504	0.552	0.534	0.539	0.575	0.583	0.591
All ABE	0.442	0.484	0.466	0.485	0.496	0.483	0.510
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.576
Other ABE	0.436	0.474	0.460	0.484	0.490	0.472	0.501

Table B.1: Complete comparison of Linear Combination approach under the Equal Weighting Strategy (LC1) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	LC2
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.490
Angina Pectoris, Variant	0.238	0.583	0.238	0.222	0.292	0.229	0.229
Angina, Unstable	0.725	0.870	0.772	0.783	0.870	0.783	0.870
Aortic Coarctation	0.816	0.964	0.742	0.742	0.890	0.862	0.816
Aortic Subvalvular Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Aortic Valve Insufficiency	0.528	0.633	0.633	0.633	0.739	0.739	0.633
Aortic Valve Stenosis	0.517	0.387	0.517	0.565	0.452	0.387	0.517
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.572
Atrial Fibrillation	0.452	0.710	0.387	0.646	0.581	0.646	0.581
Atrial Flutter	0.641	0.881	0.641	0.851	0.774	0.881	0.881
Bradycardia	0.477	0.573	0.477	0.382	0.573	0.668	0.668
Bundle-Branch Block	0.826	0.708	0.826	0.826	0.708	0.788	0.500
Carcinoid Heart Disease	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cardiac Output, Low	0.076	0.069	0.091	0.071	0.074	0.000	0.078
Cardiac Tamponade	0.573	0.668	0.477	0.668	0.477	0.668	0.668
Cardiomyopathy, Congestive	0.372	0.558	0.491	0.512	0.465	0.512	0.558
Cardiomyopathy, Hypertrophic	0.422	0.528	0.317	0.211	0.528	0.486	0.422
Cardiomyopathy, Restrictive	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Chagas Cardiomyopathy	0.450	0.750	0.250	0.750	0.750	0.750	0.750
Cor Triatriatum	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Coronary Aneurysm	0.464	0.155	0.619	0.619	0.619	0.155	0.500
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.400
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.572
Coronary Thrombosis	0.377	0.445	0.415	0.453	0.377	0.415	0.453
Coronary Vasospasm	0.171	0.367	0.183	0.367	0.550	0.514	0.550
Coronary Vessel Anomalies	0.464	0.583	0.774	0.774	0.929	0.619	0.729
Double Outlet Right Ventricle	0.000	0.000	0.625	0.000	0.000	0.000	0.000
Ductus Arteriosus, Patent	0.875	0.875	0.875	0.583	0.875	0.875	0.500
Ebstein's Anomaly	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Eisenmenger Complex	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Endocarditis	0.183	0.367	0.183	0.183	0.171	0.367	0.183
Endocarditis, Bacterial	0.621	0.552	0.552	0.737	0.621	0.760	0.621
Endomyocardial Fibrosis	0.292	0.000	0.000	0.292	0.000	0.292	0.292
Extrasystole	0.268	0.268	0.402	0.134	0.381	0.134	0.268
Heart Aneurysm	0.310	0.155	0.464	0.438	0.292	0.155	0.310
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.629
Heart Block	0.422	0.292	0.422	0.528	0.422	0.528	0.528
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.678
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.310
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.586
Heart Murmurs	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Neoplasms	0.536	0.402	0.536	0.268	0.402	0.381	0.536
Heart Rupture	0.550	0.367	0.367	0.550	0.550	0.367	0.367
Heart Rupture, Post-Infarction	0.500	0.250	0.500	0.500	0.500	0.250	0.500
Heart Septal Defects	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Septal Defects, Atrial	0.583	0.583	0.583	0.583	0.800	0.583	0.583
To be cont'd ...							

Table cont'd ...							
Heart Septal Defects, Ventricular	0.438	0.464	0.464	0.438	0.464	0.464	0.464
Heart Valve Diseases	0.207	0.483	0.276	0.414	0.469	0.483	0.483
Kearns Syndrome	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Long QT Syndrome	0.583	0.583	0.875	0.583	0.875	0.583	0.583
Mitral Valve Insufficiency	0.414	0.483	0.402	0.621	0.483	0.621	0.621
Mitral Valve Prolapse	0.292	0.583	0.583	0.583	0.583	0.583	0.583
Mitral Valve Stenosis	0.515	0.707	0.572	0.686	0.629	0.743	0.780
Myocardial Diseases	0.345	0.402	0.276	0.276	0.276	0.207	0.345
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.818
Myocarditis	0.155	0.310	0.464	0.464	0.464	0.464	0.438
Pericardial Effusion	0.550	0.550	0.550	0.367	0.550	0.550	0.550
Pericarditis	0.183	0.183	0.183	0.550	0.550	0.367	0.550
Pericarditis, Constrictive	0.550	0.733	0.550	0.514	0.550	0.367	0.550
Pulmonary Heart Disease	0.583	0.875	0.583	0.583	0.583	0.583	0.583
Pulmonary Valve Insufficiency	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Pulmonary Valve Stenosis	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Rheumatic Heart Disease	0.250	0.267	0.267	0.292	0.292	0.000	0.583
Shock, Cardiogenic	0.590	0.590	0.590	0.590	0.472	0.590	0.590
Sick Sinus Syndrome	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.684
Tachycardia, Atrioventricular Nodal Reentry	0.225	0.196	0.225	0.225	0.225	0.225	0.225
Tachycardia, Ectopic Atrial	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Ectopic Junctional	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Paroxysmal	0.619	0.464	0.464	0.619	0.619	0.619	0.619
Tachycardia, Supraventricular	0.586	0.668	0.537	0.716	0.683	0.716	0.683
Tetralogy of Fallot	0.583	0.583	0.583	0.533	0.583	0.583	0.583
Transposition of Great Vessels	0.225	0.417	0.450	0.450	0.450	0.225	0.450
Tricuspid Valve Insufficiency	0.183	0.343	0.171	0.367	0.183	0.367	0.183
Tricuspid Valve Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Truncus Arteriosus, Persistent	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Ventricular Fibrillation	0.297	0.445	0.371	0.431	0.297	0.431	0.445
Ventricular Outflow Obstruction	0.417	0.417	0.417	0.833	0.417	0.417	0.417
Wolff-Parkinson-White Syndrome	0.675	0.625	0.675	0.675	0.675	0.675	0.675
Myocardial Reperfusion Injury	0.489	0.489	0.534	0.400	0.489	0.578	0.489
Torsades de Pointes	0.750	0.750	0.750	0.750	0.750	0.750	0.750
All MBE	0.504	0.552	0.534	0.539	0.575	0.583	0.582
All ABE	0.442	0.484	0.466	0.485	0.496	0.483	0.501
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.574
Other ABE	0.436	0.474	0.460	0.484	0.490	0.472	0.490

Table B.2: Complete comparison of Linear Combination approach under the Weighting Strategy Based On Utility Measure (LC2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	LC3
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.490
Angina Pectoris, Variant	0.238	0.583	0.238	0.222	0.292	0.229	0.229
Angina, Unstable	0.725	0.870	0.772	0.783	0.870	0.783	0.870
Aortic Coarctation	0.816	0.964	0.742	0.742	0.890	0.862	0.816
Aortic Subvalvular Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Aortic Valve Insufficiency	0.528	0.633	0.633	0.633	0.739	0.739	0.633
Aortic Valve Stenosis	0.517	0.387	0.517	0.565	0.452	0.387	0.517
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.595
Atrial Fibrillation	0.452	0.710	0.387	0.646	0.581	0.646	0.581
Atrial Flutter	0.641	0.881	0.641	0.851	0.774	0.881	0.801
Bradycardia	0.477	0.573	0.477	0.382	0.573	0.668	0.668
Bundle-Branch Block	0.826	0.708	0.826	0.826	0.708	0.788	0.826
Carcinoid Heart Disease	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cardiac Output, Low	0.076	0.069	0.091	0.071	0.074	0.000	0.081
Cardiac Tamponade	0.573	0.668	0.477	0.668	0.477	0.668	0.668
Cardiomyopathy, Congestive	0.372	0.558	0.491	0.512	0.465	0.512	0.558
Cardiomyopathy, Hypertrophic	0.422	0.528	0.317	0.211	0.528	0.486	0.422
Cardiomyopathy, Restrictive	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Chagas Cardiomyopathy	0.450	0.750	0.250	0.750	0.750	0.750	0.750
Cor Triatriatum	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Coronary Aneurysm	0.464	0.155	0.619	0.619	0.619	0.155	0.464
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.385
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.582
Coronary Thrombosis	0.377	0.445	0.415	0.453	0.377	0.415	0.453
Coronary Vasospasm	0.171	0.367	0.183	0.367	0.550	0.514	0.550
Coronary Vessel Anomalies	0.464	0.583	0.774	0.774	0.929	0.619	0.875
Double Outlet Right Ventricle	0.000	0.000	0.625	0.000	0.000	0.000	0.000
Ductus Arteriosus, Patent	0.875	0.875	0.875	0.583	0.875	0.875	0.875
Ebstein's Anomaly	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Eisenmenger Complex	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Endocarditis	0.183	0.367	0.183	0.183	0.171	0.367	0.183
Endocarditis, Bacterial	0.621	0.552	0.552	0.737	0.621	0.760	0.621
Endomyocardial Fibrosis	0.292	0.000	0.000	0.292	0.000	0.292	0.292
Extrasystole	0.268	0.268	0.402	0.134	0.381	0.134	0.381
Heart Aneurysm	0.310	0.155	0.464	0.438	0.292	0.155	0.417
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.638
Heart Block	0.422	0.292	0.422	0.528	0.422	0.528	0.486
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.657
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.310
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.586
Heart Murmurs	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Neoplasms	0.536	0.402	0.536	0.268	0.402	0.381	0.536
Heart Rupture	0.550	0.367	0.367	0.550	0.550	0.367	0.550
Heart Rupture, Post-Infarction	0.500	0.250	0.500	0.500	0.500	0.250	0.500
Heart Septal Defects	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Septal Defects, Atrial	0.583	0.583	0.583	0.583	0.800	0.583	0.583
To be cont'd ...							

Table cont'd ...							
Heart Septal Defects, Ventricular	0.438	0.464	0.464	0.438	0.464	0.464	0.464
Heart Valve Diseases	0.207	0.483	0.276	0.414	0.469	0.483	0.483
Kearns Syndrome	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Long QT Syndrome	0.583	0.583	0.875	0.583	0.875	0.583	0.800
Mitral Valve Insufficiency	0.414	0.483	0.402	0.621	0.483	0.621	0.621
Mitral Valve Prolapse	0.292	0.583	0.583	0.583	0.583	0.583	0.583
Mitral Valve Stenosis	0.515	0.707	0.572	0.686	0.629	0.743	0.629
Myocardial Diseases	0.345	0.402	0.276	0.276	0.276	0.207	0.402
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.809
Myocarditis	0.155	0.310	0.464	0.464	0.464	0.464	0.464
Pericardial Effusion	0.550	0.550	0.550	0.367	0.550	0.550	0.550
Pericarditis	0.183	0.183	0.183	0.550	0.550	0.367	0.550
Pericarditis, Constrictive	0.550	0.733	0.550	0.514	0.550	0.367	0.550
Pulmonary Heart Disease	0.583	0.875	0.583	0.583	0.583	0.583	0.583
Pulmonary Valve Insufficiency	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Pulmonary Valve Stenosis	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Rheumatic Heart Disease	0.250	0.267	0.267	0.292	0.292	0.000	0.267
Shock, Cardiogenic	0.590	0.590	0.590	0.590	0.472	0.590	0.590
Sick Sinus Syndrome	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.684
Tachycardia, Atrioventricular Nodal Reentry	0.225	0.196	0.225	0.225	0.225	0.225	0.225
Tachycardia, Ectopic Atrial	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Ectopic Junctional	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Paroxysmal	0.619	0.464	0.464	0.619	0.619	0.619	0.619
Tachycardia, Supraventricular	0.586	0.668	0.537	0.716	0.683	0.716	0.683
Tetralogy of Fallot	0.583	0.583	0.583	0.533	0.583	0.583	0.533
Transposition of Great Vessels	0.225	0.417	0.450	0.450	0.450	0.225	0.450
Tricuspid Valve Insufficiency	0.183	0.343	0.171	0.367	0.183	0.367	0.183
Tricuspid Valve Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Truncus Arteriosus, Persistent	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Ventricular Fibrillation	0.297	0.445	0.371	0.431	0.297	0.431	0.445
Ventricular Outflow Obstruction	0.417	0.417	0.417	0.833	0.417	0.417	0.417
Wolff-Parkinson-White Syndrome	0.675	0.625	0.675	0.675	0.675	0.675	0.675
Myocardial Reperfusion Injury	0.489	0.489	0.534	0.400	0.489	0.578	0.445
Torsades de Pointes	0.750	0.750	0.750	0.750	0.750	0.750	0.750
All MBE	0.504	0.552	0.534	0.539	0.575	0.583	0.591
All ABE	0.442	0.484	0.466	0.485	0.496	0.483	0.511
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.574
Other ABE	0.436	0.474	0.460	0.484	0.490	0.472	0.502

Table B.3: Complete comparison of Linear Combination approach under the Weighting Strategy Based On Document Rank (LC3) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	MUDOF	IDEAL
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.598	0.598
Angina Pectoris, Variant	0.238	0.583	0.238	0.222	0.292	0.229	0.222	0.583
Angina, Unstable	0.725	0.870	0.772	0.783	0.870	0.783	0.783	0.870
Aortic Coarctation	0.816	0.964	0.742	0.742	0.890	0.862	0.742	0.964
Aortic Subvalvular Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Aortic Valve Insufficiency	0.528	0.633	0.633	0.633	0.739	0.739	0.739	0.739
Aortic Valve Stenosis	0.517	0.387	0.517	0.565	0.452	0.387	0.517	0.565
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.536	0.584
Atrial Fibrillation	0.452	0.710	0.387	0.646	0.581	0.646	0.646	0.710
Atrial Flutter	0.641	0.881	0.641	0.851	0.774	0.881	0.641	0.881
Bradycardia	0.477	0.573	0.477	0.382	0.573	0.668	0.382	0.668
Bundle-Branch Block	0.826	0.708	0.826	0.826	0.708	0.788	0.826	0.788
Carcinoid Heart Disease	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cardiac Output, Low	0.076	0.069	0.091	0.071	0.074	0.000	0.071	0.091
Cardiac Tamponade	0.573	0.668	0.477	0.668	0.477	0.668	0.668	0.668
Cardiomyopathy, Congestive	0.372	0.558	0.491	0.512	0.465	0.512	0.491	0.558
Cardiomyopathy, Hypertrophic	0.422	0.528	0.317	0.211	0.528	0.486	0.211	0.528
Cardiomyopathy, Restrictive	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Chagas Cardiomyopathy	0.450	0.750	0.250	0.750	0.750	0.750	0.750	0.750
Cor Triatriatum	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Coronary Aneurysm	0.464	0.155	0.619	0.619	0.619	0.155	0.619	0.619
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.356	0.445
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.502	0.565
Coronary Thrombosis	0.377	0.445	0.415	0.453	0.377	0.415	0.415	0.453
Coronary Vasospasm	0.171	0.367	0.183	0.367	0.550	0.514	0.367	0.514
Coronary Vessel Anomalies	0.464	0.583	0.774	0.774	0.929	0.619	0.774	0.929
Double Outlet Right Ventricle	0.000	0.000	0.625	0.000	0.000	0.000	0.625	0.625
Ductus Arteriosus, Patent	0.875	0.875	0.875	0.583	0.875	0.875	0.875	0.875
Ebstein's Anomaly	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Eisenmenger Complex	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Endocarditis	0.183	0.367	0.183	0.183	0.171	0.367	0.367	0.367
Endocarditis, Bacterial	0.621	0.552	0.552	0.737	0.621	0.760	0.737	0.760
Endomyocardial Fibrosis	0.292	0.000	0.000	0.292	0.000	0.292	0.292	0.292
Extrasystole	0.268	0.268	0.402	0.134	0.381	0.134	0.134	0.402
Heart Aneurysm	0.310	0.155	0.464	0.438	0.292	0.155	0.438	0.464
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.580	0.638
Heart Block	0.422	0.292	0.422	0.528	0.422	0.528	0.528	0.528
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.644	0.678
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.190	0.225
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.602	0.602
Heart Murmurs	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Neoplasms	0.536	0.402	0.536	0.268	0.402	0.381	0.402	0.536
Heart Rupture	0.550	0.367	0.367	0.550	0.550	0.367	0.367	0.550
Heart Rupture, Post-Infarction	0.500	0.250	0.500	0.500	0.500	0.250	0.500	0.500
Heart Septal Defects	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Septal Defects, Atrial	0.583	0.583	0.583	0.583	0.800	0.583	0.583	0.800
Heart Septal Defects, Ventricular	0.438	0.464	0.464	0.438	0.464	0.464	0.464	0.464
Heart Valve Diseases	0.207	0.483	0.276	0.414	0.469	0.483	0.483	0.483
Kearns Syndrome	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Long QT Syndrome	0.583	0.583	0.875	0.583	0.875	0.583	0.583	0.875
Mitral Valve Insufficiency	0.414	0.483	0.402	0.621	0.483	0.621	0.621	0.621
Mitral Valve Prolapse	0.292	0.583	0.583	0.583	0.583	0.583	0.583	0.583
Mitral Valve Stenosis	0.515	0.707	0.572	0.686	0.629	0.743	0.743	0.743
Myocardial Diseases	0.345	0.402	0.276	0.276	0.276	0.207	0.207	0.402
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.799	0.832
Myocarditis	0.155	0.310	0.464	0.464	0.464	0.464	0.464	0.464
Pericardial Effusion	0.550	0.550	0.550	0.367	0.550	0.550	0.367	0.550
Pericarditis	0.183	0.183	0.183	0.550	0.550	0.367	0.550	0.550
Pericarditis, Constrictive	0.550	0.733	0.550	0.514	0.550	0.367	0.514	0.733
To be cont'd ...								

Table cont'd ...								
Pulmonary Heart Disease	0.583	0.875	0.583	0.583	0.583	0.583	0.583	0.875
Pulmonary Valve Insufficiency	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Pulmonary Valve Stenosis	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Rheumatic Heart Disease	0.250	0.267	0.267	0.292	0.292	0.000	0.292	0.292
Shock, Cardiogenic	0.590	0.590	0.590	0.590	0.472	0.590	0.590	0.590
Sick Sinus Syndrome	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.608	0.684
Tachycardia, Atrioventricular Nodal Reentry	0.225	0.196	0.225	0.225	0.225	0.225	0.225	0.225
Tachycardia, Ectopic Atrial	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Ectopic Junctional	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Paroxysmal	0.619	0.464	0.464	0.619	0.619	0.619	0.464	0.619
Tachycardia, Supraventricular	0.586	0.668	0.537	0.716	0.683	0.716	0.716	0.716
Tetralogy of Fallot	0.583	0.583	0.583	0.533	0.583	0.583	0.533	0.583
Transposition of Great Vessels	0.225	0.417	0.450	0.450	0.450	0.225	0.450	0.450
Tricuspid Valve Insufficiency	0.183	0.343	0.171	0.367	0.183	0.367	0.367	0.367
Tricuspid Valve Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Truncus Arteriosus, Persistent	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Ventricular Fibrillation	0.297	0.445	0.371	0.431	0.297	0.431	0.371	0.445
Ventricular Outflow Obstruction	0.417	0.417	0.417	0.833	0.417	0.417	0.833	0.833
Wolff-Parkinson-White Syndrome	0.675	0.625	0.675	0.675	0.675	0.675	0.675	0.675
Myocardial Reperfusion Injury	0.489	0.489	0.534	0.400	0.489	0.578	0.534	0.578
Torsades de Pointes	0.750	0.750	0.750	0.750	0.750	0.750	0.750	0.750
All MBE	0.504	0.552	0.534	0.539	0.575	0.583	0.561	0.607
All ABE	0.442	0.484	0.466	0.485	0.496	0.483	0.501	0.560
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.542	0.585
Other ABE	0.436	0.474	0.460	0.484	0.490	0.472	0.495	0.557

Table B.4: Complete comparison of MUDOF approach (MUDOF) and the ideal combination of algorithms (IDEAL) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus.

Category	RO	WH	KNN	SVM	GISR	GISW	MUDOF2
Angina Pectoris	0.344	0.536	0.454	0.449	0.516	0.598	0.495
Angina Pectoris, Variant	0.238	0.583	0.238	0.222	0.292	0.229	0.229
Angina, Unstable	0.725	0.870	0.772	0.783	0.870	0.783	0.847
Aortic Coarctation	0.816	0.964	0.742	0.742	0.890	0.862	0.890
Aortic Subvalvular Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Aortic Valve Insufficiency	0.528	0.633	0.633	0.633	0.739	0.739	0.633
Aortic Valve Stenosis	0.517	0.387	0.517	0.565	0.452	0.387	0.517
Arrhythmia	0.541	0.575	0.536	0.432	0.584	0.572	0.580
Atrial Fibrillation	0.452	0.710	0.387	0.646	0.581	0.646	0.581
Atrial Flutter	0.641	0.881	0.641	0.851	0.774	0.881	0.881
Bradycardia	0.477	0.573	0.477	0.382	0.573	0.668	0.573
Bundle-Branch Block	0.826	0.708	0.826	0.826	0.708	0.788	0.826
Carcinoid Heart Disease	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cardiac Output, Low	0.076	0.069	0.091	0.071	0.074	0.000	0.150
Cardiac Tamponade	0.573	0.668	0.477	0.668	0.477	0.668	0.668
Cardiomyopathy, Congestive	0.372	0.558	0.491	0.512	0.465	0.512	0.558
Cardiomyopathy, Hypertrophic	0.422	0.528	0.317	0.211	0.528	0.486	0.528
Cardiomyopathy, Restrictive	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Chagas Cardiomyopathy	0.450	0.750	0.250	0.750	0.750	0.750	0.750
Cor Triatriatum	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Coronary Aneurysm	0.464	0.155	0.619	0.619	0.619	0.155	0.310
Coronary Arteriosclerosis	0.267	0.356	0.218	0.356	0.311	0.445	0.400
Coronary Disease	0.466	0.540	0.552	0.502	0.556	0.565	0.582
Coronary Thrombosis	0.377	0.445	0.415	0.453	0.377	0.415	0.453
Coronary Vasospasm	0.171	0.367	0.183	0.367	0.550	0.514	0.733
Coronary Vessel Anomalies	0.464	0.583	0.774	0.774	0.929	0.619	0.619
Double Outlet Right Ventricle	0.000	0.000	0.625	0.000	0.000	0.000	0.000
Ductus Arteriosus, Patent	0.875	0.875	0.875	0.583	0.875	0.875	0.875
Ebstein's Anomaly	0.750	0.750	0.750	0.750	0.750	0.750	0.750
Eisenmenger Complex	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Endocarditis	0.183	0.367	0.183	0.183	0.171	0.367	0.183
Endocarditis, Bacterial	0.621	0.552	0.552	0.737	0.621	0.760	0.621
Endomyocardial Fibrosis	0.292	0.000	0.000	0.292	0.000	0.292	0.292
Extrasystole	0.268	0.268	0.402	0.134	0.381	0.134	0.402
Heart Aneurysm	0.310	0.155	0.464	0.438	0.292	0.155	0.417
Heart Arrest	0.638	0.609	0.580	0.543	0.638	0.609	0.638
Heart Block	0.422	0.292	0.422	0.528	0.422	0.528	0.528
Heart Defects, Congenital	0.493	0.678	0.543	0.644	0.534	0.610	0.667
Heart Diseases	0.225	0.222	0.139	0.190	0.197	0.222	0.357
Heart Failure, Congestive	0.436	0.602	0.552	0.493	0.556	0.602	0.602
Heart Murmurs	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Neoplasms	0.536	0.402	0.536	0.268	0.402	0.381	0.536
Heart Rupture	0.550	0.367	0.367	0.550	0.550	0.367	0.550
Heart Rupture, Post-Infarction	0.500	0.250	0.500	0.500	0.500	0.250	0.500
Heart Septal Defects	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Heart Septal Defects, Atrial	0.583	0.583	0.583	0.583	0.800	0.583	0.875
Heart Septal Defects, Ventricular	0.438	0.464	0.464	0.438	0.464	0.464	0.464
Heart Valve Diseases	0.207	0.483	0.276	0.414	0.469	0.483	0.483
Kearns Syndrome	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Long QT Syndrome	0.583	0.583	0.875	0.583	0.875	0.583	0.583
Mitral Valve Insufficiency	0.414	0.483	0.402	0.621	0.483	0.621	0.690
Mitral Valve Prolapse	0.292	0.583	0.583	0.583	0.583	0.583	0.583
Mitral Valve Stenosis	0.515	0.707	0.572	0.686	0.629	0.743	0.686
Myocardial Diseases	0.345	0.402	0.276	0.276	0.276	0.207	0.414
Myocardial Infarction	0.781	0.811	0.789	0.750	0.832	0.799	0.812
Myocarditis	0.155	0.310	0.464	0.464	0.464	0.464	0.464
Pericardial Effusion	0.550	0.550	0.550	0.367	0.550	0.550	0.550
Pericarditis	0.183	0.183	0.183	0.550	0.550	0.367	0.550
Pericarditis, Constrictive	0.550	0.733	0.550	0.514	0.550	0.367	0.733
To be cont'd ...							

Table cont'd ...							
Pulmonary Heart Disease	0.583	0.875	0.583	0.583	0.583	0.583	0.583
Pulmonary Valve Insufficiency	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Pulmonary Valve Stenosis	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Rheumatic Heart Disease	0.250	0.267	0.267	0.292	0.292	0.000	0.292
Shock, Cardiogenic	0.590	0.590	0.590	0.590	0.472	0.590	0.590
Sick Sinus Syndrome	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Tachycardia	0.684	0.582	0.684	0.608	0.700	0.633	0.684
Tachycardia, Atrioventricular Nodal Reentry	0.225	0.196	0.225	0.225	0.225	0.225	0.225
Tachycardia, Ectopic Atrial	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Ectopic Junctional	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Tachycardia, Paroxysmal	0.619	0.464	0.464	0.619	0.619	0.619	0.619
Tachycardia, Supraventricular	0.586	0.668	0.537	0.716	0.683	0.716	0.683
Tetralogy of Fallot	0.583	0.583	0.583	0.533	0.583	0.583	0.533
Transposition of Great Vessels	0.225	0.417	0.450	0.450	0.450	0.225	0.675
Tricuspid Valve Insufficiency	0.183	0.343	0.171	0.367	0.183	0.367	0.367
Tricuspid Valve Stenosis	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Truncus Arteriosus, Persistent	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Ventricular Fibrillation	0.297	0.445	0.371	0.431	0.297	0.431	0.445
Ventricular Outflow Obstruction	0.417	0.417	0.417	0.833	0.417	0.417	0.417
Wolff-Parkinson-White Syndrome	0.675	0.625	0.675	0.675	0.675	0.675	0.675
Myocardial Reperfusion Injury	0.489	0.489	0.534	0.400	0.489	0.578	0.489
Torsades de Pointes	0.750	0.750	0.750	0.750	0.750	0.750	0.750
All MBE	0.504	0.552	0.534	0.539	0.575	0.583	0.597
All ABE	0.442	0.484	0.466	0.485	0.496	0.483	0.523
Top 10 ABE	0.488	0.551	0.505	0.497	0.542	0.566	0.582
Other ABE	0.436	0.474	0.460	0.484	0.490	0.472	0.515

Table B.5: Complete comparison of MUDOF2 approach (MUDOF2) with existing component classification algorithms based on macro-averaged recall and precision break-even point measures for the OHSUMED corpus.

Bibliography

[1] C. Apte, F. Damerau, and S.Weiss. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems*, 12(3):233–251, 1993.

[2] B. T. Bartell, G. W. Cottrell, and R. K. Belew. Automatic combination of multiple ranked retrieval systems. In *Proceedings of the 17th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 173–181, 1994.

[3] N. J. Belkin, P. Kantor, E. A. Fox, and J. A. Shaw. Combining the evidence of multiple query representations for information retrieval. *Information Processing and Management*, 31(3):269–448, 1995.

[4] P. K. Chan and S. J. Stolfo. Comparative evaluation of voting and meta-learning on partitioned data. In *Proceedings of the Twelfth International Conference on Machine Learning (ICML'95)*, pages 90–98, 1995.

[5] W. W. Cohen and Y. Singer. Context-sensitive learning methods for text categorization. In *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 307–315, 1996.

[6] H. Drucker, D. Wu, and V. N. Vapnik. Support vector machines for spam categorization. *IEEE Transactions On Neural Networks*, 10(5):1048–1054, 1999.

[7] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In *Proceedings*

of the Seventh International Conference on Information and Knowledge Management, pages 148–155, 1998.

[8] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML'98)*, 1998.

[9] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[10] E. H. Han, G. Karypis, and V. Kumar. Text categorization using weight adjusted k-nearest neighbor classification. In *Proceedings of the 5th Pacific-Asia Conference, PAKDD 2001*, pages 53–65, 2001.

[11] T. K. Ho. Complexity of classification problems and comparative advantages of combined classifiers. In *Proceedings of the First International Workshop on Multiple Classifier Systems (MCS 2000)*, pages 97–106, 2000.

[12] D. A Hull, J. O. Pedersen, and H. Schutze. Method combination for document filtering. In *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 279–287, 1996.

[13] R. D. Iyer, D. D. Lewis, R. E. Schapire, Y. Singer, and A. Singhal. Boosting for document routing. In *Proceedings of the Ninth International Conference On Information Knowledge Management (CIKM 2000)*, pages 70–77, 2000.

[14] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European Conference on Machine Learning (ECML'98)*, pages 137–142, 1998.

[15] G. Karypis and E. H. Han. Fast supervised dimensionality reduction algorithm with applications to document categorization & retrieval.

In *Proceedings of the Ninth International Conference On Information Knowledge Management (CIKM 2000)*, pages 12–19, 2000.

[16] J. Kivinen and M. K. Warmuth. Averaging expert predictions. In *European Conference On Computational Learning Theory*, pages 153–167, 1999.

[17] D. Koller and M. Sahami. Hierarchically classifying documents using very few words. In *Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97)*, pages 170–178, 1997.

[18] S. Kumar, J. Ghosh, and M. Crawford. A hierarchical multiclassifier system for hyperspectral data analysis. In *Proceedings of the First International Workshop on Multiple Classifier Systems (MCS 2000)*, pages 270–279, 2000.

[19] K. Y. Lai and W. Lam. Automatic textual document categorization using multiple similarity-based models. In *Proceedings of the First SIAM International Conference on DATA MINING (SDM 2001)*, 2001.

[20] K. Y. Lai and W. Lam. Meta-learning models for automatic textual document categorization. In *Proceedings of the 5th Pacific-Asia Conference (PAKDD 2001)*, pages 78–89, 2001.

[21] W. Lam and C. Y. Ho. Using a generalized instance set for automatic text categorization. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 81–89, 1998.

[22] W. Lam and K. Y. Lai. A meta-learning approach for text categorization. In *Proceedings of the 24th International ACM SIGIR Conference on Research and Development in Information Retrieval (To appear)*, 2001.

[23] W. Lam, K. F. Low, and C. Y. Ho. Using a Bayesian network induction approach for text categorization. In *Proceedings of the Fifteenth Inter-*

national Joint Conference on Artificial Intelligence, (IJCAI), Nagoya, Japan, pages 745–750, 1997.

[24] L. S. Larkey and W. B. Croft. Combining classifiers in text categorization. In *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 289–297, 1996.

[25] J. H. Lee. Analyses of multiple evidence combination. In *Proceedings of the 20th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 267–276, 1997.

[26] D. D. Lewis. Evaluating text categorization. In *The Speech and Natural Language Workshop, Asilomar*, pages 289–297, 1991.

[27] D. D. Lewis, R. E. Schapore, J. P. Call, and R. Papka. Training algorithms for linear text classifiers. In *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 298–306, 1996.

[28] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[29] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[30] G. Salton M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[31] D. Meretakis, D. Fragoudis, H. Lu, and S. Likothanassis. Scalable association-based text classification. In *Proceedings of the Ninth International Conference On Information Knowledge Management (CIKM 2000)*, pages 5–11, 2000.

[32] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the Ninth International Conference On Information Knowledge Management (CIKM 2000)*, pages 86–93, 2000.

[33] K. Nigam and A. K. McCallum. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39:103–134, 2000.

[34] G. Salton. A theory of indexing. *Regional Conference Series in Applied Mathematics No. 18, Society for Industrial and Applied Mathematics, Philadelphia, PA*, pages 351–372, 1973.

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

[36] G. Salton and C. S. Yang. On the specification of term values in automatic indexing. *Journal of Documentation*, pages 351–372, 1973.

[37] R. Schapire and Y. Singer. BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39(213):135–168, 2000.

[38] F. Sebastiani, A. Sperduti, and N. Valdambrini. An improved boosting algorithm and its application to text categorization. In *Proceedings of the Ninth International Conference On Information Knowledge Management (CIKM 2000)*, pages 78–85, 2000.

[39] S. Y. Sohn. Meta analysis of classification algorithms for pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1137–1144, November 1999.

[40] A. H. Tan. Predictive self-organizing networks for text categorization. In *Proceedings of the 5th Pacific-Asia Conference (PAKDD 2001)*, pages 66–77, 2001.

[41] K. M. Ting and I. H. Witten. Stacked generalization: when does it work? In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, pages 866–871, 1997.

[42] V. Vapnic. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.

[43] K. Wang, S. Zhou, and S. C. Liew. Building hierarchical classifiers using class proximity. In *Proceedings of the 25th VLDB Conference*, pages 363–374, 1999.

[44] Y. Yang. Expert network: Effective and efficient learning from human decisions in text categorization and retrieval. In *Proceedings of the 17th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 13–22, 1994.

[45] Y. Yang, T. Ault, and T. Pierce. Combining multiple learning strategies for effective cross validation. In *Proceedings of the International Conference on Machine Learning (ICML 2000)*, pages 1167–1174, 2000.

[46] Y. Yang, T. Ault, T. Pierce, and C. W. Lattimer. Improving text categorization methods for event tracking. In *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 65–72, 2000.

[47] Y. Yang and C. D. Chute. An example-based mapping method for text categorization and retrieval. *ACM Transactions on Information Systems*, 12(3):252–277, 1994.

[48] Y. Yang and X. Liu. A re-examination of text categorization methods. In *Proceedings of the 22th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 42–49, 1999.

[49] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the International Conference on Machine Learning (ICML'97)*, pages 412–420, 1997.