Information Retrieval: A Framework for Recommending Text-based Classification Algorithms

By

Hany Saleeb

Submitted in Partial Fulfillment
Of the Requirements for the degree of
Doctor of Professional Studies
In Computing

at

School of Computer Science and Information Systems

June 2002

ProQuest Information and Learning Company
300 North Zeeb Road
PO Box 1346
Ann Arbor, MI 48106-1346

Research Signature Approval Page

We hereby certify that this research, submitted by Hany Saleeb, satisfies the research requirements for the degree of Doctor of Professional Studies and has been approved.

_____-_____
Fred Grossman Date
Chairperson of Research Committee

_____-_____
Frances Gustavson Date
Research Committee Member

_____-_____
Allen Stix Date
Research Committee Member

School of Computer Science and Information Systems
Pace University 2002

An Abstract

Information Retrieval:
A Framework for
Recommending
Text-based
Classification Algorithms

by

Hany Saleeb

Submitted in Partial Fulfillment
Of the Requirements for the degree of
Doctor of Professional Studies
In Computing

June 2002

Classification is one of the central issues in information retrieval systems dealing with text data. The need for effective approaches has been dramatically increased due to the advent of the World Wide Web and massive digital libraries. Effective methods are invaluable for the exploration of information repositories with the aim to discover similarities between groups of text-based documents.

One goal of this thesis is the development of tools for supporting users of machine learning and data mining algorithms in the area of text classification. While the interest in such technology is growing rapidly, tools are still limited to end-users who are not experts. This is due to the fact that machine learning systems are difficult to design and their number keeps increasing. As a result, system designers are faced with two major research problems: algorithmic model selection and model combination, i.e., (a) selecting the most suitable model/algorithm to use on a given application, and (b) integrating this with useful and effective transformations of the data. Traditionally, these problems are resolved by trial-and-error or through consultation of experts. The first solution is time consuming and unreliable. The second solution is expensive and biased by the expert's own prejudices and preferences. This thesis develops a meta-model framework system called the Regression Model Framework (RMF) that supports system designers with model selection and method combination. RMF uses statistical regression analysis to combine prior meta-knowledge with meta-level learning.

The second major goal of this thesis is to investigate how text classification is performed on the Web. A great deal of text-based documents are available on the Internet and in corporate intranets, and categorizing them into useful semantic categories is a rewarding

and challenging research problem. However, current approaches to text categorization on the Web mostly concentrate on simple representation schemes that are based on word occurrence and word frequency. The structural information that is inherent to documents on the Web is usually neglected. In analyzing Web documents, the relative importance of hypertext tags is investigated in order to ascertain their relative importance in predicting the relevance of unknown documents.

Acknowledgments

I want to thank my advisor, Fred Grossman. Throughout my studies, he was the source of many appealing research ideas, many of which subsequently bore fruit, some of which I had the opportunity to realize and extend. “Opportunity” is a good word to use in describing Fred whose advisory style involves treating his advisees as colleagues, spreading before them a wealth of interests, directions and inviting them to find their niche. At times, I was overwhelmed. His true strength as an advisor did not become completely clear to me until my thesis was in the home stretch, when in spite of a schedule that would hobble most people, he carefully reviewed and criticized each of the chapters. The result is a better, more precise presentation.

I would like to include the many faculty members whose discussions were invaluable. I thank Fran Gustavson and Allen Stix as my committee members. Ron Frank was always knowledgeable about a vast number of topics such as databases, software engineering and statistics. His frankness and simplicity were always much appreciated.

An important contribution to my success is the fellowship with other students. I had some good ones: Leslie Beckford, Chris Iervolino, Suman Kalia, Jonathan Law, Stephen Parshley and Pat Wong. Many hours were spent over Instant Messenger and the phone discussing our progress and frustrations. Their insight and perspectives was always refreshing. Our conversations about work, family and success enlightened me.

Table of Contents

1 INTRODUCTION 1

1.1 TEXT PROCESSING 2
1.1.1 Text Classification Tasks *3*
1.1.1.1 Text Retrieval 3
1.1.1.2 Text Categorization 4
1.1.1.3 Text Routing 5
1.1.1.4 Term Categorization 6
1.1.1.5 Document Clustering 6
1.1.1.6 Term Clustering 7
1.1.1.7 Latent Indexing 7
1.1.2 Text Classification Research Focus *8*
1.1.2.1 Algorithm Selection 10
1.1.2.2 Multi-Criteria Algorithm Selection 12
1.2 WORLD WIDE WEB 13
1.2.1 Taxonomy of Web Mining Techniques *14*
1.2.1.1 Web Content Mining 16
1.2.1.2 Web Structure Mining 18
1.2.1.3 Web Usage Mining 18
1.2.2 Thesis Focus on Web classification *20*
1.3 OUTLINE FOR THE DISSERTATION 20

2 BASICS OF TEXT CLASSIFICATION 22

2.1 INTRODUCTION 22
2.2 PREPROCESSING 25
2.3 INDEXING 26
2.3.1 Boolean Weighting *28*
2.3.2 Word Frequency Weighting *29*
2.3.3 Term and Document Frequency Weighting *29*
2.3.4 Entropy Weighting *30*
2.4 DIMENSIONALITY REDUCTION 30
2.4.1 Feature Selection *31*
2.4.1.1 Document Frequency Thresholding 31
2.4.1.2 Information Gain 31
2.4.1.3 χ^2 statistic 32
2.4.2 Re-parameterization *33*
2.4.2.1 Singular Value Decomposition 34
2.5 TRAINING AND TESTING 35
2.6 CATEGORIZATION ALGORITHMS 37
2.6.1 Rocchio's Algorithm *37*
2.6.2 k-Nearest Neighbor *39*
2.6.3 Naïve Bayes *41*
2.6.4 Decision Trees *43*
2.6.4.1 CART 44
2.6.4.2 C4.5 46
2.6.5 Support Vector Machines *46*
2.6.6 Voted Classification *49*
2.6.7 Maximum Entropy *50*
2.6.7.1 Constraints and Features 51
2.6.8 Document Clustering *52*
2.6.9 Neural Networks *53*
2.7 PERFORMANCE MEASURES 54
2.7.1 Multiple Binary Classification Tasks *54*
2.7.1.1 Micro and Macro-averaging 57

2.7.1.2 Breakeven point ... 59
2.7.1.3 F-measure ... 59
2.7.2 Multi-class and Multi-label Classification ... 61
2.8 SUMMARY ... 62

3 BAG OF WORDS AND TERM CATEGORIZATION ... 64

3.1 INTRODUCTION ... 64
3.2 SETUP ... 65
3.3 DIRECT COMPARISON AMONG FIVE CLASSIFIERS ... 68
Newsgroup ... 73
3.4 TERM CATEGORIZATION ... 73
3.4.1 Proposed term categorization word selection algorithm ... 78
3.4.2 Experimentation on Adjacent Bigrams ... 80
3.4.3 Experimentation on Nonadjacent Bigrams ... 87
3.4.4 Significance of Results ... 92
3.4.5 Experimentation of bigrams with different classifiers ... 93
3.5 NUMBER OF FEATURES ... 95
3.6 BENCHMARKS ... 96
3.6.1 Which classifier performs best? ... 100
3.6.1.1 Classifier Model Expertise ... 102
3.6.1.1.1 Classification Quality ... 103
3.6.1.1.2 Scalability ... 104
3.7 SUMMARY ... 106

4 THE REGRESSION MODEL FRAMEWORK ... 108

4.1 OVERVIEW ... 108
4.2 FRAMEWORK META-KNOWLEDGE ... 112
4.3 FRAMEWORK ALGORITHM ... 114
4.3.1 Ranking Methodology ... 118
4.4 APPLICATION SPECIFIC USER PREFERENCES ... 119
4.5 RMF SYSTEM FLOW ... 124
4.5.1 RMF Recommending a Classifier ... 126
4.5.2 RMF Adding a New Algorithm ... 127
4.6 SUMMARY ... 130

5 REGRESSION MODEL FRAMEWORK EXPERIMENTATION ... 131

5.1 STATISTICAL CONFIDENCE ... 131
5.2 REUTERS DATASET ... 133
5.3 APPLICATION: PATENTS ... 137
5.3.1 Patent Dataset Experimentation ... 139
5.4 APPLICATION: CORPORATE INTRANETS ... 144
5.4.1 Corporate Web Site Dataset Experimentation ... 145
5.4.2 Retrieval Time ... 154
5.5 ALGORITHM GROUPINGS ... 157
5.5.1 Strengths and Weakness of Symbolic Learning Methods ... 158
5.5.2 Strengths and Weakness of Case Based Methods ... 161
5.5.3 Strengths and Weakness of Neural Nets ... 162
5.5.4 Strengths and Weakness of Statistical Methods ... 166
5.6 META-MODEL LIMITATIONS/CONSTRAINTS ... 167
5.6.1 Incremental Learning ... 167
5.6.2 Original Dataset ... 169
5.7 SUMMARY ... 169

6 HIERARCHIAL STRUCTURE OF DOCUMENTS ... 171

6.1 INTRODUCTION ... 172
6.1.1 Discovering resources on the Internet ... 172

6.1.2 Web Information Retrieval *174*
6.2 MOTIVATION 176
6.3 STRUCTURAL ALGORITHM 179
6.4 VOTING METHODS 181
6.4.1 Experimental Setup *183*
6.4 RESULTS 185
6.4.1 Page Accuracy *185*
6.4.2 Link Accuracy *186*
6.4.3 Recall and Precision *187*
6.4.4 Comparison to full text classifier *189*
6.5 A LINK-BASED MODEL 190
6.5.1 Hyper Information *192*
6.5.2 Single Links *194*
6.5.3 Fading Function *197*
6.5.4 Multiple Links *199*
6.5.5 The General Case *199*
6.6 SUMMARY 201

7 CONCLUSION AND FUTURE WORK 202

7.1 FUTURE WORK 204
7.1.1 Restricting Attention to Relevant Datasets *205*
7.1.2 Natural Language Processing *206*
7.1.3 Ranking Classifiers *207*
7.1.4 Incremental Classification *207*
7.1.5 Multimedia *208*
7.1.6 Adding new algorithms using sampled datasets *208*
7.1.6.1 Categories of Sampling Strategies 209
7.1.6.2 Random sampling 209
7.1.6.3 Domain Sampling 210
7.1.6.4 Probability Sampling 210
7.1.6.5 Adaptive vs. Non-adaptive Sampling 211
7.2 SUMMARY 212

8 REFERENCES 214

APPENDIX 1: AN EXAMPLE OF A PATENT 221

APPENDIX 2: GLOSSARY 229

Index of Tables

TABLE 1 : WORD MATRIX AFTER INDEXING 2 DOCUMENTS 27
TABLE 2: STATISTICS IN INFORMATION RETRIEVAL 55
TABLE 3: SOME BAG-OF-WORDS RESEARCH 65
TABLE 4: MAKE UP OF THE NEWSGROUPS USED FOR EXPERIMENTATION 67
TABLE 5: NAIVES BAYES CLASSIFICATION 70
TABLE 6: MOST USEFUL WORDS FOR MODELING 70
TABLE 7: NAÏVE BAYES MIS-CLASSIFICATION BETWEEN 2 NEWSGROUPS 71
TABLE 8: NB, SVM, TFIDF AND MAXENT CLASSIFICATIONS PRECISION 72
TABLE 9: SUPPORT VECTOR MACHINE MIS-CLASSIFICATION BETWEEN 2 NEWSGROUPS 73
TABLE 10: MAXIMUM ENTROPY MISCLASSIFICATION BETWEEN 2 NEWSGROUPS 73
TABLE 11: NUMBER OF DOCUMENTS IN EACH CATEGORY 81
TABLE 12: NUMBER OF BIGRAMS EXTENDED 82
TABLE 13: TOP 10 BIGRAMS FOR EACH CATEGORY 83
TABLE 14: INFORMATION GAIN IN BIGRAMS 84
TABLE 15: BREAKEVEN POINTS FOR BIGRAMS 85
TABLE 16: F_1 FOR BIGRAMS 85
TABLE 17: RECALL/PRECISION FOR BIGRAMS 86
TABLE 18: PERFORMANCE AS A METRIC OF FEATURE PROXIMITY 88
TABLE 19: SIGNIFICANCE TESTS ON N-GRAM FEATURE SETS 93
TABLE 20: PRECISION AND RECALL OF CLASSIFIERS VERSUS N-GRAM TOKENIZATION 94
TABLE 21 : PRECISION AS THE NUMBER OF FEATURES INCREASES 96
TABLE 22 : CHARACTERISTICS OF THE FIVE REUTERS BENCHMARKS 98
TABLE 23: COMPARATIVE RESULTS OF PAST REUTERS BENCHMARKS 99
TABLE 24: MICRO-AVERAGING ON REUTERS-21578 103
TABLE 25: F1 SCORES FOR NB, KNN AND SVM ON THE TOP 10 CATEGORIES OF REUTERS-21578 104
TABLE 26: TRAINING TIME FOR FEATURE SET SIZE OF 50 105
TABLE 27: TRAINING TIME FOR FEATURE SET SIZE OF 100 105
TABLE 28: TRAINING TIME AS A FUNCTION OF THE NUMBER OF FEATURES 106
TABLE 29 : SAMPLE COEFFICIENTS FOR PRECISION REGRESSION FUNCTIONS 116
TABLE 30 : ALGORITHM RANKING EXAMPLE 123
TABLE 31: REGRESSION FRAMEWORK EXPERIMENT ON TWO DATASETS 134
TABLE 32: NUMBER OF TRAINING/TESTING DOCUMENTS IN REUTERS-21578 TOP 10 135
TABLE 33: REGRESSION FRAMEWORK EXPERIMENT ON REUTERS-21578 DATASET 136
TABLE 34: PATENT DATASET USING PRECISION ONLY 140
TABLE 35 : CATEGORIZATION WITH SYNONYM LISTS 142
TABLE 36: PATENT DATASET USING RECALL ONLY 143
TABLE 37: DOCUMENTS IN COMPANY A'S TEN CATEGORIES 146
TABLE 38: PREDICTED WEB SITE PRECISION INTERVALS AT THE 90% CONFIDENCE LEVEL 148
TABLE 39: COMPARISON OF PREDICTED ALGORITHM VERSUS TRUE BEST 148
TABLE 40: VALIDATING CLASSIFIERS WITH INTEGRATED WEB SITE 150
TABLE 41 : REGRESSION MODEL FRAMEWORK: RETRIEVAL TIME 155
TABLE 42: RETRIEVAL TIME FOR THE PRODUCTS CATEGORY 156
TABLE 43: CLASS DISTRIBUTION OF LINKS AND PAGES 184
TABLE 44: ACCURACIES FOR CLASSIFYING THE 1050 PAGES 185
TABLE 45: ACCURACIES FOR CLASSIFYING 5803 LINKS 187
TABLE 46: RECALL AND PRECISION FOR PAGE PREDICTORS 188
TABLE 47: FULL-TEXT VERSUS LINK CLASSIFIERS 189
TABLE 48: ACCURACY RESULTS USING FEATURE SUBSET ON A FULL-TEXT CLASSIFIER 190

Index of Figures

FIGURE 1: TAXONOMY OF WEB MINING 15
FIGURE 2: INFORMATION RETRIEVAL SYSTEM LEVEL I
FIGURE 3: PRE-PROCESSING EXAMPLE OF A DOCUMENT 26
FIGURE 4: TRAINING AND TESTING COMPONENTS 36
FIGURE 5: K-NEAREST NEIGHBOR EXAMPLE 40
FIGURE 6: A SIMPLE CAUSALITY NETWORK 41
FIGURE 7: DECISION TREE RECURSIVE PARTITIONING 43
FIGURE 8: A LINEAR SVM SEPERATOR 47
FIGURE 9: A NON-LINEAR SVM SEPERATOR 47
FIGURE 10: SUPPORT VECTOR MACHINES EXAMPLES USING MULTI-ORDERED FUNCTIONS 48
FIGURE 11: AN EXAMPLE OF A REUTERS DOCUMENT 68
FIGURE 12: ALGORITHM FOR GENERATING N-GRAM TOKENIZED WORDS 78
FIGURE 13: FRAMEWORK SYSTEM LEVEL DESIGN 110
FIGURE 14: OVERALL RMF DESIGN 111
FIGURE 15: REGRESSION MODEL FRAMEWORK BASED ON PRECISION 117
FIGURE 16: INTERACTION WITH MATLAB 117
FIGURE 17: REGRESSION MODEL FRAMEWORK BASED ON MULTIPLE CRITERIA 121
FIGURE 18: RMF SYSTEM FLOW 125
FIGURE 19: AN RMF USER ASKING FOR A CLASSIFIER RECOMMENDATION 126
FIGURE 20: BLOCK DIAGRAM OF ADDING AN ALGORITHM 128
FIGURE 21: AN RMF USER ADDS A NEW ALGORITHM 129
FIGURE 22: USER INTERACTION FOR NEWSGROUP 133
FIGURE 23: WEB SITE CLASSIFICATION MECHANISM 147
FIGURE 24: NUMBER OF HOSTS THROUGH THE YEARS 173
FIGURE 25: NUMBER OF WEB SITES OVER THE YEARS 173
FIGURE 26: VOTING AMONG LINKED PAGES 182
FIGURE 27: ONE WEB PAGE LINKING TO ANOTHER WEB PAGE 194
FIGURE 28: A WEB PAGE RECURSIVELY BEING LINKED FROM OTHER WEB PAGES 195
FIGURE 29: A WEB PAGE LINKING TO ANOTHER WEB PAGE 197
FIGURE 30: A WEB PAGE RECURSIVELY BEING LINKED FROM OTHER WEB PAGES 198
FIGURE 31: A WEB PAGE RECURSIVELY BEING LINKED FROM OTHER WEB PAGES AT DEPTH 1 199
FIGURE 32: A WEB PAGE RECURSIVELY BEING LINKED FROM OTHER WEB PAGES AT DEPTH 2 200

The problems that we have today were created at a particular level of thinking.
We can not solve these problems at the same level of thinking.
-Albert Einstein

Chapter 1

Introduction

As the volume of information available on the Internet and corporate intranets continues to increase rapidly, there is a growing need for tools helping people find, filter and manage these resources. Text classification, the assignment of text documents to one or more predefined categories based on their content, is an important component to many information management tasks. These include real-time filtering of email or files into folder hierarchies, topic identification to support topic specific processing operations, structured search and/or browsing, and finding documents that match long term standing interests.

In many contexts, trained professionals are employed to categorize new items. This process is very time consuming and costly, thus limiting its applicability [15]. Consequently, there is an increasing interest in developing technologies for automatic text categorization.

The growing availability of information sources especially large, non-homogenous distributed sources like the World Wide Web has produced a need for effective ways to

filter information. This research addresses the area of automatic text classification which integrates text processing with data mining, machine learning and statistics.

1.1 Text Processing

By almost any measure, the amount of information being produced is growing faster than the ability of information consumers to find, digest, and use this information [14] [57]. One response has been to publish information in computer-accessible form rather than by traditional media such as paper, film, audio tape or video tapes. Businesses and other organizations store an increasing amount of their internally generated information in computer-accessible form. A small but increasing proportion of both technical and everyday correspondence and conversation occurs, and is recorded, by electronic mail and voice mail, adding to the opportunities and problems created by information growth.

Textual data is difficult to effectively understand because the relationship between its sequence of words and its content is less clear than, for example, numerical [44]. Textual data includes technical articles, memos, manuals, electronic mail, books, newspapers, magazines, journals and many other forms of text. In addition, it is desirable to access other forms of data including speech, images and video through textual annotations.

Content-based text processing tasks can be divided into two broad groups. First, text classification involves the assigning of documents or parts of documents to one or more of a number of categories. Second, text understanding involves more complete access to

the content of documents such as extracting formatted data, answering questions, and summarization or abstracting. This research addresses the first group.

1.1.1 Text Classification Tasks

Classification is a common term in information retrieval, applied statistics and psychology referring to processes of grouping entities. Text classification, therefore, is an appropriate term to subsume a number of information retrieval tasks that are usually considered distinct, but which all involve grouping of textual entities. Text classification is made up of seven sub-tasks [52][53]: text retrieval, text categorization, text routing, term categorization, document clustering, term clustering and latent indexing.

1.1.1.1 Text Retrieval

Text retrieval is the computer selection of a subset of a document database in whole or summary form for an end user, usually in response to a user request. One view of a text retrieval system is that it sorts documents into two classes: documents that will be displayed to the user and those that will not. Many advanced text retrieval systems not only select documents but also attempt to order displayed documents by importance. These systems can be viewed as measuring the degree of membership.

The text retrieval process is made up of four main phases [52]:

1. *Indexing*: Raw documents must be converted into expressions in a keyword text representation. These expressions are sometimes called features and must have a structure usable by a text retrieval software.
2. *Query formulation*: The user or external system expresses a request that can be interpreted by the information retrieval software.
3. *Comparison*: The system must implicitly or explicitly compare the user query to the stored documents and make a classification decision about which documents to retrieve and in what order.
4. *Feedback*: An initial retrieval rarely results in exactly the documents desired by a user. Several iterations of modifying the query are usually needed to achieve the desired results.

1.1.1.2 Text Categorization

Text categorization is the classification of documents with respect to a set of one or more categories. Each category is associated with a single concept or idea such as sports, science or history. The most common application of text categorization is in indexing documents for text retrieval, i.e., in producing document representatives. Manual assignment of subject categories to documents is a widely used form of text representation. Users can mention these subject categories in their requests, possibly enabling a more compact and effective query to be formed. However, manual assignment of categories requires considerable human labor and expense. This research aims to improve manual indexing with automated text classification in order to reduce these costs.

Another application of text categorization is within text understanding systems. Categorization may be used to filter out documents or parts of documents that are unlikely to contain extractable data, without incurring the costs of more complex natural language processing [26]. For example, a news information retrieval system processes

texts in many subject areas. Thus, categorization may be used to route stories to category specific topics.

As with text retrieval, a category may be binary (a document either is or is not a member of a category) or graded (a document can have a degree of membership in the category). Binary assignments have been used in most applications. When multiple categories are used, it may be the case that each document is assigned only one category. On the other hand, categories may be assigned independently, with each document falling into all, some or no categories.

Text categorization is a major focus of the research.

1.1.1.3 Text Routing

Text routing, also known as selective dissemination of information, or text filtering, combines aspects of text retrieval and text categorization. Like text categorization, a text routing system processes documents in real time and assigns them to zero or more classes. However, like text retrieval, each class is typically associated with the information needs of one or a small group of users [24]. Each user or user group can typically add, remove or modify the standing queries or user profiles associated with their needs. A user, for example, may set up a profile to filter on a single topic such as sports or even more specifically baseball. There may or may not be relationships among the user profiles and profiles may or may not be under end user control.

1.1.1.4 Term Categorization

Term categorization is similar to text categorization, in that pieces of text are assigned to predefined categories. The difference is the size of the pieces of text. Where text categorization deals with full documents or relatively large portions of documents, term categorization is the assignment of categories to words or small fragments of text. For example, in text categorization a document may be categorized by single words such as 'politics' or 'sports.' In term categorization, phrases of words may be used instead like 'international politics' or 'men's sports.' While there is some blurring of this task into text categorization, the techniques applied are different enough to consider this a distinct task.

Term categorization is addressed in this research by assessing the relative importance of the use of terms in improving categorization accuracy.

1.1.1.5 Document Clustering

Document clustering is the automated generation of categories of documents, usually based on some similarity measure between documents, as well as a definition of what characteristics groups of documents should have. Document clustering has been suggested both as a means to speed up physical access to stored documents such as

Yahoo's pre-defined categories, and as a text representation for improving the effectiveness of text retrieval.

The objectives of creating document clusters are to

- Reduce the overhead of searches. Whole clusters can be stored and retrieved together.
- Provide for a visual representation of the information space
- Expand the retrieval of relevant items: if one item in a cluster is relevant, the others probably are too.

1.1.1.6 Term Clustering

Term clustering is similar to document clustering except that individual words or small fragments consisting of closely connected words are formed into groups. It attempts to group words together to form concepts usually using tools such as a dictionary or thesaurus. Term clustering has been investigated for producing better text representations to support text retrieval, so far without much success. It has also been used as a method for studying word usage and producing information useful to natural language processing

1.1.1.7 Latent Indexing

Latent indexing is related to both term clustering and document clustering. It uses factor analysis or related techniques to transform one representation of a collection of documents into a new representation with desirable mathematical properties. Both the original indexing terms and the original documents are re-expressed in terms of this new

representation.

1.1.2 Text Classification Research Focus

Within the area of classification, this research develops and evaluates classification algorithms. This includes investigating term categorization techniques as well as performance of various text categorization algorithms. Text classification includes many sub-tasks including text categorization and term categorization. Within the context of this research, text categorization and text classification are used interchangeably.

In classification, an objective is to assign one of a set of pre-defined categories to an unknown document. A simple example of a classification problem is the evaluation of topics for a digital library. Given a research paper, a system chooses the most appropriate related topic that defines the subject of the paper. For example, a paper on the social security system in the United States can be placed in the politics topic while another dealing with a sorting algorithm can be categorized as part of the computer science subject. The classification algorithms search a space of subjects looking for one that resembles as well as governs the decisions for those cases. In a digital library, a simple rule based algorithm could analyze previous documents for vocabulary that can be used to predict the appropriate subject matter. For example, if a paper includes terminology such as ‘computer’, ‘algorithm’, ‘efficiency’ or ‘program’ then it may be classified as a computer system research paper.

Presently, the number of different categorization algorithms is large and contains contributions from many research areas. The performance of each varies widely depending on the application domain and the criteria used in assessing them. When facing a new problem, most analysts select a familiar one according to their particular experience and preference.

The characterization of the domains of applicability of algorithms has always been a central issue in the field of statistics. There are a number of drawbacks. First, a major concern of past research is that the applicability of algorithms for text categorization has only recently been given the same attention. Furthermore, the nature of these algorithms is such that comparison and characterization of algorithms is hard to achieve. This research compares various text categorization algorithms on the same dataset and measures important system performance metrics. One of the problems identified in previous work is that some of the measures used to characterize the given domain take longer to calculate than running some of the algorithms [59].

A second drawback with previous work [27][39][16] on selecting algorithms is that it concentrates on selecting the best single overall algorithm. This research argues that efforts with categorization algorithms should not be used to identify a single best overall candidate in all domains; different algorithms will perform well depending on the dataset and specific application domain requirements. This research proposes a general framework to recommend an appropriate algorithm given an application domain and system behavior requirements.

1.1.2.1 Algorithm Selection

There are many different classification algorithms that evolved from different areas in statistics, machine learning and neural networks. Thus, it is becoming challenging for a data analyst to keep up with the progress, making it difficult to select the best (or appropriate) algorithm. This problem is made more difficult by the fact that system designers who are not data analysts want to have access to classification algorithms to assist them in their decision making.

Ideally it would be optimal if it were possible to identify the single best algorithm which could be used in all situations. However, the No Free Lunch [83] theorem states that:

> "…if algorithm A outperforms algorithm B on some cost function, then loosely speaking there must exist exactly as many other functions where B outperforms A."

A brute force approach to this problem is to try all categorization algorithms on the dataset and then select the one with the best results. In practice this is not feasible in most applications because there are too many algorithms to try, some of which may be quite slow. The problem is exacerbated when dealing with large amounts of data as is common in knowledge discovery. Another problem with this approach is that the system designer and data analyst must know how to use all the algorithms.

All this implies that, given an unknown dataset, a recommendation should be given beforehand concerning which algorithm should be used. Some researchers [13] [15] describe algorithm selection as an exploratory process, highly dependent on the analyst's knowledge of the algorithms and of the problem domain, thus lying somewhere between science and art. The same authors recognize the importance of developing tools that assist the system designer in the process of selecting the best algorithm for a given problem or at least a well performing one. The need for such methods has frequently been recognized as an important issue in the fields of machine learning [58] and knowledge discovery [30]. In response to the need, this research develops a methodology to aid system designers and analysts in choosing an appropriate algorithm given their system requirements and needs.

Most previous research focused on developing, analyzing and improving algorithms for classification and other data analysis tasks, but relatively less work has been devoted to the problem of algorithm selection based on past performance. Some of the earlier approaches [15] were based on meta-knowledge concerning the performance of algorithms. This knowledge can be either analytical, empirical or both. Meta-knowledge is knowledge about a system that aids in decision making. In the analysis of the results of project StatLog [59], knowledge of both types is presented. The meta knowledge of Balabanovic [8] was of experimental origin. As used in this research, the objective of meta-knowledge is to capture certain relationships between the measured dataset

characteristics (such as the number of attributes, number of cases, skew, etc.) and the relative performance of the algorithms.

This research develops a statistical regression-based model to predict algorithms' suitability. This model is called the Regression Model Framework (RMF) and uses meta-knowledge to predict the errors of individual algorithms with a high degree of success. Furthermore, classification accuracy is not the only performance criteria used because users may have other system requirements. Thus, precision accuracy, recall, and training time, etc. are possible performance metrics. Using the Regression Model Framework, the effectiveness of predicting the best algorithm is investigated.

1.1.2.2 Multi-Criteria Algorithm Selection

Many applications require not only a single criterion but several of them. Thus, the research undertaken in this research must propose a framework such that multiple criteria can be considered. Using several criteria in evaluation of classification algorithms is a more difficult task than using just one criterion. One way to proceed is to combine them into a new measure [30]. However, problems arise:

1. Criteria may not be easily quantifiable
2. How to combine all the criteria into one measure
3. Acceptable limits on all criteria.

The first problem is related with the fact that some system designers may compare algorithms using non-traditional criteria such as success, understandability or novelty. These measures are not easily quantifiable and quite subjective. An obvious example is

novelty; different experts have certainly gathered different expertise. For the domain expertise of the Regression Framework within this research such subjective criteria are not included at all.

To solve the problem of how to combine different criteria, one may use linear or non-linear combinations of criteria. The approach taken in this research is a weighted linear combination of the criteria because the variables under consideration have a linear relationship.

The last problem is the relative importance of individual criteria or acceptable limits. Each criteria is given a weight, and a ranking is performed. For example, precision accuracy may be twice as important as training time. Of course, defining the minimum accuracy allowed is a more difficult problem to solve. How much accuracy is an application willing to give up to obtain half the training time? Little work has been done by researchers in the area of machine learning or knowledge discovery concerning this issue [15][30].

1.2 World Wide Web

The Internet contains large amounts of unstructured text-based documents, thus making it an excellent test bed for text categorization research. In a recent Asilomar Report [7] on the future of database research, it is predicted that within ten years, the majority of information will be available on the Internet. Of course, the Internet encompasses all