Large-Scale Web Page Classification

by

Sathi T Marath

Submitted in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

at

Dalhousie University

Halifax, Nova Scotia

November 2010

DALHOUSIE UNIVERSITY

FACULTY OF COMPUTER SCIENCE

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "Large-Scale Web Page Classification" by Sathi T Marath in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Dated: 9 November,2010

External Examiner: ________________________________

Research Supervisor: ________________________________

Examining Committee: ________________________________

____

Departmental Representative: ________________________________

DALHOUSIE UNIVERSITY

DATE: 9 November, 2010

AUTHOR: Sathi T Marath

TITLE: Large-Scale Web Page Classification

DEPARTMENT OR SCHOOL: Faculty of Computer Science

DEGREE: PhD CONVOCATION: May YEAR: 2011

Signature of Author

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Web page classification is the process of assigning predefined categories to web pages. Empirical evaluations of classifiers such as Support Vector Machines (SVMs), k-Nearest Neighbor (k-NN), and Naïve Bayes (NB), have shown that these algorithms are effective in classifying small segments of web directories. The effectiveness of these algorithms, however, has not been thoroughly investigated on large-scale web page classification of such popular web directories as Yahoo! and LookSmart. Such web directories have hundreds of thousands of categories, deep hierarchies, spindle category and document distributions over the hierarchies, and skewed category distribution over the documents. These statistical properties indicate class imbalance and rarity within the dataset.

In hierarchical datasets similar to web directories, expanding the content of each category using the web pages of the child categories helps to decrease the degree of rarity. This process, however, results in the localized overabundance of positive instances especially in the upper level categories of the hierarchy. The class imbalance, rarity and the localized overabundance of positive instances make applying classification algorithms to web directories very difficult and the problem has not been thoroughly studied. To our knowledge, the maximum number of categories ever previously classified on web taxonomies is 246,279 categories of Yahoo! directory using hierarchical SVMs leading to a Macro-F1 of 12% only.

We designed a unified framework for the content based classification of imbalanced hierarchical datasets. The complete Yahoo! web directory of 639,671 categories and 4,140,629 web pages is used to setup the experiments. In a hierarchical dataset, the prior probability distribution of the subcategories indicates the presence or absence of class imbalance, rarity and the overabundance of positive instances within the dataset. Based on the prior probability distribution and associated machine learning issues, we partitioned the subcategories of Yahoo! web directory into five mutually exclusive groups. The effectiveness of different data level, algorithmic and architectural solutions to the associated machine learning issues is explored. Later, the best performing classification technologies for a particular prior probability distribution have been identified and integrated into the Yahoo! Web directory classification model. The methodology is evaluated using a DMOZ subset of 17,217 categories and 130,594 web pages and we statistically proved that the methodology of this research works equally well on large and small dataset.

The average classifier performance in terms of macro-averaged F1-Measure achieved in this research for Yahoo! web directory and DMOZ subset is 81.02% and 84.85% respectively.

LIST OF ABBREVIATIONS AND SYMBOLS USED

SVM	Support Vector Machine
K-NN	K-Nearest Neighbor
NB	Naïve Bayes
ODP	Open Directory Project
ACENet	Atlantic Computational Excellence Network
HPC	High Performance Computing
HTML	Hypertext Markup Language
PCA	Principal Component Analysis
LSI	Latent Semantic Indexing
SVD	Singular Value Decomposition
LLSF	Linear Least Square Fit
LR	Logistic Regression
TP Rate	True Positive Rate
FP Rate	False Positive Rate
ROC	Receiver Operator Characteristics
AUC	Area Under Roc
QF	Quality Factor
DCT	Distributed Computing Toolbox
MPI	Message Passing Interface
ODP	Open Directory Project

ACKNOWLEDGEMENTS

This thesis represents a milestone in my academic life. At this point 1 would like to acknowledge help, encouragement and support offered by several teachers, colleagues, friends and family members that helped me to complete this successfully.

First and foremost, I would like to express my deep sense of gratitude to my supervisor Dr. Michael Shepherd, Professor of Faculty of Computer Science, Dalhousie University, Canada for providing valuable guidance throughout this research work. He has been a constant point of reference and support during my academic pursuit in Dalhousie University. Regular discussions with him during the course of this research helped me to clarify my thoughts and to translate them into coherent ideas. I greatly appreciate all his contributions of time, expert guidance, and funding to make my Ph.D. experience productive and stimulating.

I would like to express my sincere thanks to the research committee members Dr. Evangelos Milios and Dr. Malcolm Heywood for taking academic interest in this research as well as offering timely help in the form of comments and suggestions.

I am glad to render my deep sense of gratitude to Dr. Jack Duffy, for his valuable suggestions at several occasions during the course of my research and expert guidance in the statistical analysis of dataset and results.

I am thankful to Dr. Ross Dickson, Research Consultant of Atlantic Computational Excellence Network (ACEnet), for providing excellent technical support.

I am extending cordial appreciations to my brother-in-law, Dr. Ravindran Pulyassary, on behalf of his valuable suggestions in the area of High Performance Computing which indeed helped to improve the cluster utilization and speed of the web directory classification model.

I am thankful to the members of Web Information Filtering Lab for numerous ideas and suggestions during the course of this research.

I wish to express my sincere thanks to my family members, colleagues, and friends for their keen interest, continuous support and cooperation during the period of my research work.

CHAPTER 1: INTRODUCTION

Over the past decade, web users have been witnessing an exponential growth in the number of web pages accessible through popular search engines. Organizing the large volume of web information in a well-ordered and accurate way is critical for using it as an information resource. One way of accomplishing this in a meaningful way requires web page classification. Web page classification addresses the problem of assigning predefined categories to the web pages by means of supervised learning. This inductive process automatically builds a model by learning over a set of previously classified web pages. The learned model is then used to classify new web pages. This technology integrates Information Retrieval, Data mining, Machine Learning and Natural Language Processing.

Numerous classifiers proposed and used for machine learning can be applied for web page classification. These include Support Vector Machines (SVMs), k-Nearest Neighbor (k-NN), and Naïve Bayes (NB) classifiers. Empirical evaluations of these algorithms on selected small segments of popular web directories have shown that most of these methods are effective in web page classification (Chen, 2000; Sebastiani, 2002; Yang Y., 1999). However, the effectiveness of these algorithms on very large web taxonomies like the Yahoo! directory and Open Directory Project (ODP) is not thoroughly investigated. Web taxonomies like the Yahoo! directory and the Open Directory Project have hundreds of thousands of categories and millions of web pages. The sheer volume of categories and web pages makes large-scale web page classification an inevitable component for web directories and search engines.

In contrast to the traditional benchmark datasets, web directories generally have complex statistical properties. This makes large-scale hierarchical web page classification significantly different from the traditional text classification and web page classification with limited categories and documents. Web directories usually have more categories and documents in the middle of the hierarchy than at either the upper or the lower levels of the hierarchy. This spindled distribution is an indication of the class imbalance within the dataset. The class imbalance problem is a relatively new research area, which emerged during the growth of machine learning from its embryonic state to an applied technology. In an imbalanced dataset, almost all examples belong to one class, while far fewer examples represent the other class. When a machine learning algorithm is exposed to an imbalanced dataset, standard classifiers tend to focus on the large classes and ignore the small classes. In addition, popular evaluation measures such as accuracy place more weight on the common classes than on rare classes. Thus, the performance with respect to small classes is difficult to assess (Japkowicz, N.,& Stephen, S, 2002; Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006).

Another distinguishing attribute of web directories is the skewed category distribution over the web pages. The number of web pages assigned to the categories follows the power law distribution (Liu,T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma,W., 2004). The skewed category distribution and power law distribution on the number of web pages indicates that most categories have very few labeled web pages. This indicates rarity within the dataset. Data level or algorithmic treatments are necessary to learn the rare categories of the web directory.

In web taxonomies similar to Yahoo!, the assignment of a web page into a category will not automatically grant this assignment to its parent categories or vice versa. The recursive assignment of the web pages of a category into its parent category helps to decrease the degree of rarity within web taxonomies. This process, however, results in the localized over-abundance of positive instances especially in the upper level categories of the hierarchy. When classifying categories with very large numbers of positive training instances, it is crucial to assess whether the classifier trained with a very large dataset is better than the one trained with a small subset of data. In theory, classifier performance should not be reduced when trained on a large dataset. However, classifiers using large dataset for training may not always be better, and may be slightly worse due to the much larger solution space.

The wide variation in the content and quality of the web page is another challenge of large-scale web page classification. Most of the categorization algorithms assume that the training data is of good quality**.** Web pages, however, have highly variable size and different tag formats along with noise content such as advertisement banners and navigation bars. Thus, compared to other text datasets, web pages lack homogeneity and regularity. Furthermore, a huge number of distinct words exist in the pages including proper words and misspelled words. Thus, an intelligent preprocessing of the web pages is necessary (John, M.P., 2000).

Web page classification is an inductive procedure that automatically builds a model by learning over a set of previously classified web pages. Hence, the degree of agreement on the category of a web page among a group of raters, also known as inter-rater reliability, is critical for web page classification. Unfortunately, the inter-rater reliability of popular web directories is not well studied.

Liu et al. (Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004) evaluated the performance of flat and hierarchical SVMs on a 246,279 category subset of the Yahoo! directory. To our knowledge, this is the maximum number of categories ever previously classified on web taxonomies. In their research, hierarchical SVMs lead to a Micro-F1 of 24% and a Macro-F1 of 12%. The authors conclude that in terms of effectiveness neither flat nor hierarchical SVMs can fulfill the classification needs of very large-scale taxonomies. The skewed distribution of the large web directories like Yahoo! with many extremely rare categories makes SVM performance ineffective. Their research, however, completely overlooked the machine learning aspects and solutions to the class imbalance and absolute rarity. This may be the root cause for poor SVM classifier performance.

Different statistical properties of web taxonomies question whether the existing web page classification technologies can perform well on large and imbalanced web taxonomies. The difficulties in applying classification algorithms to very large web taxonomies are not thoroughly studied. Previous web page categorization research on a few common categories or selected small segments of web taxonomies could not preserve the original characteristics of the web taxonomy as a whole. Hence, the observations from earlier studies do not take a broad view of this area.

This research investigates the development of a unified framework for the content based classification of imbalanced hierarchical datasets such as web directories. In an imbalanced dataset like Yahoo! web directory, the prior probability distribution of a category indicates the presence or absence class imbalance, alone or together with absolute rarity or large-sample learning issues due to the overabundance of positive instances. Based on the prior probability distribution and associated machine learning issues, we partitioned the subcategories of Yahoo! web directory into 5 mutually exclusive groups. The effectiveness of different data level, algorithmic and architectural solutions to these machine learning issues is explored. Later, the best performing classification technologies for a particular prior probability distribution have been identified and used to design a content based classification model for complete Yahoo! web directory of 639,671 categories and 4,140,629 web pages. Afterward, the methodology is evaluated using a DMOZ subset of 17,217 categories and 130,594 web pages and we statistically proved that the methodology of this research works equally well on large and small datasets.

A thorough review to evaluate the breadth and depth of the issues pertaining to web page classification technology is discussed in Chapter 2. A typical web page classification process consists of steps such as feature selection, feature extraction, classifier design, and finally performance evaluation. Numerous feature selection methods, feature extraction methods, and classifiers have been proposed and were used for the web page classification problem. However, previous web page categorization research on a few common categories or selected small segments of web taxonomies could not preserve the original characteristics of the web taxonomy as a whole. Hence, the observations from earlier studies do not take a broad view of this area.

Different data level, algorithmic, and architectural solutions to the over-abundance of positive instances, class imbalance and rarity problem associated with classification research have been proposed and were used by the machine learning community. The effectiveness of these approaches in large-scale web page classification is critically analyzed in Chapter 3.

The methodology of this research is discussed in Chapter 4. This includes multiple machine learning models to classify an imbalanced dataset with localized over-abundance of positive instances, rarity and class imbalance. In Chapters 5, 6 and 7, these machine learning models combined with popular feature selection methods such as Information Gain, Document Frequency and popular classifiers such as Perceptron, Support Vector Machine and Maximum Entropy Classifiers, have been examined and their relative merits and demerits are critically analyzed. Later, a Yahoo! web directory classification model is designed using the best performing classification technologies. The Yahoo! web directory classification model is discussed in Chapter 8.

Whether the methodology of this research works equally well on large and small dataset is examined in Chapter 9. A DMOZ subset of 17,217 categories is used to set up the experiments. At the time of our crawling in October, 2009, there were 602,410 categories and 4,519,050 web pages in the topmost 14 levels of the DMOZ web directory. The category distribution of the DMOZ web directory with hierarchy depth is similar to that of Yahoo! web directory. Evaluation of the methodology using a DMOZ subset of 17,217 categories is discussed in Chapter 9.

Chapter 10 is the conclusion and future research. The breadth and depth of the issues pertaining to the large-scale web page classification technology is studied in this research. The average classifier performance in terms of macro-averaged F1-Measure achieved in this research for Yahoo! web directory and DMOZ subset is 81.02% and 84.85% respectively. To our knowledge, the maximum number of categories ever previously classified on web taxonomies is 246,279 categories of Yahoo! directory. In their research, hierarchical SVMs lead to a Macro-F1 of 12% only. Similarly the highest average F1-Measure reported for DMOZ subset is 35.37%. In these research works, the hierarchical classifier evaluation procedure they followed to calculate the reported Macro-F1 measure is not clear.

There are a few areas in large-scale web page classification that need more investigation. The impact of class imbalance on the popular feature selection measures is not examined in this research. However, preliminary studies are conducted and we conclude that the statistical feature selection method such as Information Gain is not optimal for the classification of very large web directories.

At this point, extreme rarity prevents training individual classifiers for categories with fewer than 10 labeled web pages. We cannot expect any statistical learner to perform well on such rare categories. In this research, the classifiers of the parent categories have been used to classify these categories. The advantage of merging extreme rare categories with the parent categories is applicable to the hierarchical dataset only. Around 70% of the categories of the popular web directories are extremely rare with fewer than 10 labeled instances. A better alternative to categorize these categories will complement many real-world flat and hierarchical classification problems including text classification, medical dataset classification and intrusion detection.

CHAPTER 2: REVIEW OF POPULAR WEB PAGE CLASSIFICATION TECHNOLOGIES

2.1 Introduction

Web page classification is essential to many tasks in Web Information Retrieval, such as maintaining web directories and focused crawling. Compared to traditional text classification datasets such as the Reuter's corpus, web pages generally have variable size and different tag formats along with noise content such as advertisement banners and navigation bars. The irregular nature of the web pages and their exponential growth in number make web page classification an inexhaustible challenge. Different web page classification technologies from machine learning and Information Retrieval have been proposed and their relative merits on classifying the new web pages have been experimentally evaluated. This chapter reviews these technologies. This includes different web page preprocessing techniques, accepted dimensionality reduction methods, popular web page classifiers and popular evaluation measures.

The overall goals of the review are to address the following queries:

1. Why intelligent preprocessing of the web pages is required prior to the classification and how it can be achieved?
2. What are the best feature reduction and feature extraction methods?
3. What learning algorithm is most suitable for web page classification?
4. What are the limitations of earlier web page classification research?
5. Why is large-scale web page classification needed?

Web pages, compared to traditional text classification datasets, are highly irregular in nature due to the variable size, different tag formats and noise content such as advertisement banners. Hence, an intelligent preprocessing of the web pages before application of the classification algorithm is necessary. Different web page preprocessing methods that have been proposed and used by earlier web page categorization research are reviewed in Section 2.2. After preprocessing, web pages are represented as multi-dimensional vectors, where each dimension encodes a single feature of the web pages. Different web page representation methods are discussed in Section 2.3. If all the features are used to represent a candidate web page, the total dimension of the vectors will be very high. This results in high time and space complexity for the machine learning algorithm. Various dimensionality reduction functions, from information theory and linear algebra, have been proposed and their relative merits have been experimentally evaluated. These functions are divided into feature selection and feature extraction functions based on the nature of features chosen. Detailed reviews of different feature selection and feature extraction functions are discussed in Sections 2.4 and 2.5.

Dimensionality reduction is also beneficial to reduce the problems of classifier over fitting. Over fitting is the phenomenon where a classifier is tuned to the training data, rather than being generalized from essential characteristics of the training data to classify a new web page (Sebastiani F., 1999). After features have been selected to form concise representations of the web pages, classification algorithms are applied to train the classifier. Various classification algorithms proven efficient for web page classification are reviewed in Section 2.6. Different classifier evaluation metrics are discussed in Section 2.7. A summary of this literature review is provided in Section 2.8.

2.2 Web Page Preprocessing

Web pages are very dynamic in structure with variable size, different tag formats and noise contents. The tag format as well as the quantity of textual content within the different tags varies widely resulting in an inconsistency in the information across the different segments of a web page. Intelligent preprocessing of the web pages is needed prior to the classification. Web page preprocessing integrates different approaches to identify the concise portion of the web page and its cleaning to remove the noise and less informative terms such as stop words. Various web page preprocessing approaches using the HTML structure and hypertext structure have been studied and their effectiveness in the context of web page classification has been evaluated.

2.2.1 Web Page Preprocessing using the HTML Structure of the Web Pages

Web pages, in contrast to a traditional text dataset, encapsulate the structural information in the form of HTML tags. This structural information could be useful to enhance the informative segment(s) identification. For example, the HTML structure TITLE gives information about the content of the web page. BODY, META TITLE and META DESCRIPTION are other excellent textual information sources of the web page. However, using different intermediary tools, very short web pages with little text information and more non-text based contents can be designed. The HTML structure of these types of web pages will not convey much information about their content. With the help of the linked pages, attempts were made to represent these types of web pages effectively.

2.2.2 Web Page Preprocessing using the Hypertext Nature of the Web Pages

Web page preprocessing using the hypertext nature of web pages assumes that a link is created only if there is a relationship between the contents of the original web page and the connected web page. However, a crude and raw combination of the local full text and the text in the linked web pages may not help feature selection and classification. This is due to the hypertext regularities. The presence or absence of the hypertext regularities such as Meta data, Pre-classified, Co-referencing, Encyclopedia, and None can significantly influence the relationship between linked web pages and the original web page (Yang Y., 1999). Different studies using IBM patent web pages, Yahoo! corpus (Chakrabati. S., Dom, B., & Indyk, P., 1998) and online encyclopedia articles (Oh, H., Myaeng, S.H., & Lee, M., 2000) also agree with this observation. In these studies, increasing the feature space using the text data of the linked web pages resulted in the accuracy decrease of 6% and 24% respectively. Instead of adding the complete vocabulary, a focused upgrading of the web pages using the anchor text and text nearby the anchor text of in-linked web pages has also been studied. Even though anchor text seems to be informative, web page classification research shown that using the anchor text alone is less efficient compared to the classification using the full text (Blum A. & Mitchell. T., 1998; Glover, E. J., Tsioutsiouliklis, K., Lawrence, S., Pennock, D.M., Flake, & G.W., 2002). However, alternative web page representation using the terms from the anchor text, headings preceding the anchor text, and paragraphs where the anchor text occurs in the in-linked web pages improved the performance by 20% compared to the web page representation using local full text (Furnkranz, 1999).

The work cited in this subsection provides some insights in exploring the structural information for web page classification. However, drawing general conclusions in this area can be misleading (Yang Y, 2001). A better approach may be to perform a quantitative analysis on the dataset and identify the information rich segment(s) applicable to a majority of the web pages. The following steps remove the less informative contents of the identified segment(s):

1. Removing HTML tags.
2. Removing scripting languages such as java script
3. Removing stop words
4. Word stemming

The textual information remaining after the preprocessing (known as features) is used for the web page representation. Web page representation is the process of projecting the textual information, after preprocessing, in a meaningful way for the purpose of feature reduction and classification.

2.3 Web Page Representation

The popular web page representation for web page classification is the bag-of-words representation. In the bag-of-words representation, a web page is characterized by a vector d_i with words $t_1,t_2,...,t_M$ as the features, each of which associates with a weight w_{ij}. That is $d_i=w_{i1},w_{i2},w_{i3},.....w_{iM}$ where M is the number of indexing words and w_{ij} is the importance of term t_j in the web page d_i, often represented as the frequency. The bag-of-words representation assumes that each word in a document signifies the concept of the document. A phrase usually contains more information than a single word. Hence, the bag-of-words representation can be enriched by using word sequence. The bag-of-words representation, however, does not preserve the structural information formed by the HTML tags and the hyperlinks of the web page.

After the web page representation, the whole collection of web pages may contain hundreds of thousands of unique terms. If all the unique terms are used for representing the web pages, the dimension of the feature vectors will be enormous. For a web page categorization problem, dimensionality reduction is necessary due to the following reasons.

1. If all features are used to represent a candidate web page, the total dimension of the vectors will be very high. This results in high time and space complexity for the machine learning algorithm.
2. Dimensionality reduction is also beneficial to reduce the problems of classifier over fitting (Sebastiani F., 2002). Over fitting is the phenomenon where a classifier is tuned to the training data, rather than being generalized from essential characteristics of the training data to classify a new web page.
3. For a classification problem, a smaller feature space can give either better or as good results as a larger feature space (Tikk,D., Bansaghi,Z.,& Biro,G., 2005).

Various dimensionality reduction functions from information theory and linear algebra have been proposed and their relative merits have been evaluated. These functions can be divided into feature selection and feature extraction functions based on the nature of features chosen.

2.4 Dimensionality Reduction by Feature Selection

Two broad approaches available for dimensionality reduction by feature selection are the wrapper approach (Kohavi, R., & John, G. H., 1997; John. G., Kohavi, R., & Pfleger, K., 1994) and the filter approach (John. G., Kohavi, R., & Pfleger, K., 1994). The wrapper approach employs search through the feature subspace. Taking the neural network as an example, the wrapper approach starts training with an initial subset of features and measures the performance of the network. If the classification error is beyond the given limit, an improved feature set with more features is generated and network performance is measured. This process is repeated until the termination condition in minimal error value or total number of iterations is reached. The high time and space complexity due to the huge size of web page dataset and feature set makes the wrapper approach highly inappropriate for web page classification.

The filter approach is an alternative feature selection method more suitable to web page classification. In the filter approach, feature selection is detached from the learning algorithm and is performed as a preprocessing step prior to the machine learning. Hence, the filter algorithm does not bring additional time complexity to classification systems.

Considering the advantages of the filter approach over the wrapper approach and the suitability of the filter approach for web page classification problem, wrapper approaches are not discussed in this review.

The filter approach processes the features independently and assigns a numeric sore to the features based on some statistical criteria. The best features for the classification process are selected by fixing a predefined threshold of the assigned score. Many feature selection criteria from statistics and information theory have been studied and their relative merits on identifying the discriminating features have been evaluated**.** In a broad view, the filter approach based feature selection criteria can be divided into two sets. One set of feature selection methods, such as, Document Frequency, Mutual Information, Cross Entropy, and Odds Ratio considers the possible value of features that are present in the document. The other set of feature selection methods, such as, Information Gain and Chi-square Statistic, considers all possible values of features including those that are present in and those that are absent from a document (Yang, Y., & Pederson, J. O., 1997; Mladenic. D. & Grobelnik. M., 1998, Mladenic. D. & Grobelnik. M., 1999).

2.4.1 Comparison of Different Feature Selection Techniques

While many feature selection techniques have been proposed, a thorough evaluation of these methods over a very large feature space is not reported. However, Yang et al. (Yang, Y., & Pederson, J. O., 1997) and Mladenic et al., (Mladenic, D., & Grobelnik, M., 1998; Mladenic, D., & Grobelnik, M., 1999) did remarkable research in this area. Yang et al. (1997) evaluated the effectiveness of Information Gain, Chi-square statistics, Document frequency, and Mutual Information as feature selection methods and their relative merits on classification using k-nearest neighbor (k-NN) and Linear Least Squares Fit mapping algorithms. The Reuter's collection and the OHSUMED collection were used to set up the experiments. The Information Gain feature selection method achieved up to 98% reduction in the feature space and yielded 10% improvement in the classification accuracy. This research reported Information Gain and Chi-square statistics as more effective for feature selection as compared to Document Frequency and Mutual Information. However, considering the strong correlation among document frequency, information gain and chi-square statistics established in this research, we may conclude that, document frequency, the simplest feature selection method with lowest cost complexity, can also be reliably used in place of the computational expensive information gain and chi-square statistics. This research reported mutual information as a weak feature selection criteria and this naturally points to the inherent bias of the mutual information towards the rare features. However, in their research, removing the rare features from the feature set do not made any remarkable improvement on mutual information compared with other measures.

Mladenic et al. (Mladenic, D., & Grobelnik, M., 1998; Mladenic, D., & Grobelnik. M., 1999) evaluated the effectiveness of Odds Ratio, Cross Entropy, Information Gain and Mutual Information in association with the Naive Bayes classifier. Web pages from the Yahoo! dataset were used to set up the experiments. This research reported Odds Ratio and Cross Entropy as the two best performing feature selection methods. Mutual information showed poorer performance than Cross Entropy. The weakest feature selection method reported in this research is Information Gain, which, on the other hand is one of the best feature selection method reported by Yang et al. (1997).

The differences in evaluation results may be due to the differences in the nature of the datasets used. Mladenic et al. used the data collection from the Yahoo! directory which has an unbalanced class distribution and highly unbalanced feature distribution. The prior probability of a feature, on an unbalanced data set with few categories will be small. In this experiment, most of the features picked by information gain may be the features having a high absent feature value. Of course, the knowledge of a feature absence in a web page conveys useful information for a classification algorithm. However, a classification scheme relying on feature absence is usually more difficult and requires a larger feature space than a classification relying on feature presence.

While choosing feature selection methods, the nature of the classification algorithm and the statistical distribution of the data domain should be taken into consideration. It is already proved that a smaller feature subset can give either better or as good results as larger feature space (Tikk, D., Bansaghi, Z., & Biro, G., 2005). Studies show that feature selection methods favoring frequent features can achieve better results compared to the methods favoring rare features (Sebastiani F., 2002). The main limitation of the discussed feature selection methods is their inability to estimate the effect of co-occurrence of features. For example, two or more features considered independently may not be very effective, but may turn highly effective, when grouped together. This limitation is addressed by applying dimensionality reduction by feature extraction.

2.5 Dimensionality Reduction by Feature Extraction

Feature extraction methods produce a set of optimum synthetic features of smaller size from the original large feature set without losing any of the significant features. Several approaches have been reported and successfully tested in this area. Principal components analysis (PCA) is a popular statistical technique for reducing a multidimensional dataset to a lower dimensional space. PCA is an orthogonal linear transformation that maps the data points into a new coordinate system in such a way that the first greatest variance by any projection of the data comes to lie on the first coordinate known as first principal component, the second greatest variance on the second coordinate, and so on. The first few principle components convey the most significant aspects of the data. By keeping the first few principle components only, PCA can be used for dimensionality reduction without losing any of the characteristic features. This is an unsupervised dimension reduction method widely used in information retrieval and text data mining. However, the vectors generated by PCA are not directly connected to the original vector space. This prevents deriving meaningful interpretations from the reduced feature space (Sebastiani F., 2000). Latent semantic indexing (LSI) is another popular feature reduction technique. LSI is based on the assumption that there is a basic or concealed semantic structure in the pattern of features used across the web page corpus. Statistical techniques are used to estimate these semantic structures. LSI uses singular value decomposition (SVD), which is a technique related to eigenvector decomposition and factor analysis. (Sebastiani F., 2002).

2.5.1 Comparison of Different Feature Extraction Methods

Techniques, such as PCA and LSI, have been shown to improve the quality of the information being retrieved by capturing the latent meaning of words present in the documents. However, after applying PCA and LSI, the discrimination power of some extremely good features may be lost in the new vector space (Sebastiani F., 2002). A few earlier researches attempted to overcome this limitation by upgrading the feature space after feature extraction with a group of manually identified feature vectors that are good for classifying given categories (Zelikovitz, S., & Hirsh,H., 2000). This is not an optimal solution for large-scale classification.

2.6 Popular Web Page Classification Algorithms and Earlier Research

After the features of training web pages have been selected to form concise representations of the web pages, various classification algorithms were applied to induce the classifier. A large number of statistical learning methods have been applied to the text classification problem in recent years. Some of them are regression models (Fuhr, N., Hartmanna, S., Lustig, G., Schwantner, M.,& Tzeras, K., 1991; Yang, Y.,& Liu, X., 1999), nearest neighbor classifiers (Creecy, & Robert, H., 1992; Yang, Y.,& Liu, X., 1999), Bayesian probabilistic classifiers (Tzeras, K.,& Hartman, S., 1993; Lewis, D. D., & Ringuette, M., 1994), decision trees (Fuhr, N., Hartmanna, S., Lustig, G., Schwantner, M., & Tzeras, K., 1991; Lewis, D. D.,& Ringuette, M.,1994), inductive rule learning algorithms (Weiss, S. M., Apte, C., Damerau, F.J., Johnson, D.E., Oles, F.J., Goetz, T., & Hampp, T., 1999; Cohen W., & Singer Y., 1999; Moulinier, I., Raskinis, G., & Ganascia, J., 1996), neural networks (Wiener, E., Pedersen, J.O., & Weigend, A.S., 1995 ; Ng, H.T., Goh, W.B.,& Low, K.L., 1997) and on-line learning approaches (Cohen W., & Singer Y., 1999 ; Lewis, D.D., Schapire, R.E., Callan, J. P.,& Papka, R., 1996). Since a large number of methods and results are available, a cross-method evaluation is important to comprehend the current status of the text categorization research. The comparison of different text and web page classification methods, however, is very difficult due to the absence of a cohesive methodology for the matter-of-fact evaluation. Cross-method comparisons with a limited number of methodologies have been reported in the literature. However, these types of small-scale comparisons can either lead to highly comprehensive statements that are based on inadequate observations, or provide limited insight into a global comparison among a wide range of approaches.

The lack of a standard data collection is the main bottle-neck for cross-method comparison in text categorization research. For a given dataset, there are many possible ways to introduce inconsistent variations. For example, the popular Reuters news story corpus has multiple versions depending on difference in the training, test and evaluation set combinations. Whether the reported classifier performance on the different versions of Reuters is comparable is not clear (Yang Y., 1999). Incomparability across different evaluation measures used in individual experiments is another concern on cross-experiment evaluation (Yang Y., 1999). Lots of measures such as recall and precision, accuracy or error, Precision-Recall breakeven point or F1-Measure have been proposed and used for the classifier evaluation. Each of these measures is designed to evaluate some characteristic of the categorization. However, none of them conveys identical or comparable information. There exist some difficulties in comparing published results of text categorization methods when they are evaluated using different performance measures. In general, one should be very vigilant while comparing the published text categorization research.

Due to the aforementioned issues, a comprehensive evaluation of different classification methods is not reported. However, Yang et al. (Yang, Y., & Liu, X., 1999) did remarkable research in this area. They published an evaluation of fourteen classifiers using the Reuter's corpus. The k-Nearest Neighbor (k-NN) classifier has shown the best performance. Other top performing classifiers listed in their research were Linear Least Square Fit (LLSF) and Neural Net. Rule induction algorithms like SWAP-1, RIPPER and CHARADE, show apparently good performance. Relatively worse performance was reported for Rocchio and Naive Bayes classifiers.

In a different study conducted by Yang (1999), the robustness of SVM, linear regression (LLSF), logistic regression (LR), Neural Net, Rocchio, Prototypes, k-Nearest Neighbor (k-NN), and the Naive Bayes classifier, when applied to a dataset with skewed category distribution were evaluated. For a skewed dataset, SVM, k-NN, and LLSF significantly outperformed Neural Net and Naive Bayes classifiers.

Different studies (Sebastiani F., 2002; Yang Y., 1999; Lewis, D. D., Yang, Y., Rose, T. G., & Li, F., 2004; Liu,T., Yang, Y., Wan,H., Zeng, H., Chen,Z., & Ma,W., 2004) have shown that SVM has high training performance and low generalization error. However, SVMs when applied to an imbalanced dataset, produce a less effective classification boundary skewed to the minority class (Akbani, R., Kwek, S., & Japkowicz, N., 2004).

In general, empirical evaluations of popular classification algorithms such as SVMs, k-NN, and Naïve Bayes classifiers on selected small segments of popular web directories have shown that most of these methods are effective in web page classification (Chen, 2000; Sebastiani, 2002; Yang Y., 1999). On the other hand, available classification research works on reasonably sized subsets of popular web directories conclude that in terms of effectiveness these classification algorithms cannot fulfill the classification needs of very large-scale taxonomies (Liu,T., Yang, Y., Wan, H., Zeng, H., Chen, Z.,& Ma, W., 2004; Chen, 2000; Dumais, S.,& Chen, H., 2000; Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008). Such web directories have hundreds of thousands of categories, deep hierarchies, spindle category and document distributions over the hierarchies, and skewed category distribution over the documents. These statistical properties indicate class imbalance and rarity (very small classes) within the data set. These distribution properties make applying classification algorithms to such data sets very difficult. A comparison of earlier large-scale web page classification is summarized in Table1.

The effectiveness of SVMs while classifying very large-scale taxonomies has been studied. Yahoo!, (Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004) and LookSmart (Chen, 2000; Dumais, S., & Chen, H., 2000) datasets were used to set up the experiments.

Table 1: A Comparison of Earlier Large-Scale Web Page Classification Research

Researcher	Dataset	No.of Categories	Depth	Method	Micro -F1 (%)	Macro -F1 (%)
Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z.,& Ma, W., 2004	Yahoo!	246,279	14	Hierarchical SVM	24	12
Chen, Dumais, S.,& Chen, H., 2000	LookSmart	163	2	Hierarchical SVM		52.40
Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008	ODP	130,000	17	Statistical language model	51.8 (at 5th level)	
Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008	ODP	130,000	17	Hierarchical SVM	29.2 (at 5th level)	

To the best of our knowledge, the maximum number of web categories ever classified is not more than 246,279 categories. This was from Yahoo! web directory (Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004). In their research, even with the best classifier setting, hierarchical SVMs lead to a Micro-F1 of 24% and a Macro-F1 of 12%. This study concludes that in terms of effectiveness neither flat nor hierarchical SVMs can fulfill the classification needs of very large-scale taxonomies. The skewed distribution of large web directories like Yahoo! with many extremely rare categories makes SVMs performance ineffective. Their research, however, completely overlooked the impact of class imbalance, and absolute rarity while classifying an imbalanced dataset.

The effectiveness of hierarchical SVM while classifying the top two levels of categories of the LookSmart dataset has also been studied. This research reported a macro-average F1 measure of 57.2% for the top level 13 categories and 47.6% for the 150 second level categories (Chen, 2000; Dumais, S., & Chen, H., 2000). There is a drop in performance in going from 13 to 150 categories. Conversely, these study uses the top two levels of the LookSmart categories only and the conclusions might not generalize to the case of classifying hundreds of thousands of categories.

Xue et al. (Xue,G., Xing, D., Yang,Q., & Yu,Y., 2008) addressed the large-scale web page classification in a two phase process. In the first phase, a category-search algorithm is executed to acquire the category candidates for a given dataset. Based on the category candidates, the large scale hierarchy is pruned and classification is performed on the pruned subset of the original hierarchy. In this research, a statistical-language-model based classifier using n-gram features is used for classification. The performance of the proposed algorithm is evaluated on the Open Directory Project with over 130,000 categories**.** With this approach the Micro-F1 at the fifth level of the hierarchy is 51.8%, whereas for top-down based SVM classification algorithms the Mico-F1 at fifth level is 29.2%.

2.7 Commonly used Evaluation Metrics for Web Page Classification

Precision, recall, accuracy, and error are the popular evaluation metrics used for classification. Precision is defined as the number of documents correctly assigned to a category divided by the total number of documents assigned to that category. Recall is defined as the number of correctly assigned documents into a category divided by total number of existing documents, which should have been assigned. Precision has a similar meaning as classification accuracy. However, they are different in that precision considers only examples assigned to the category, while accuracy considers both assigned and rejected cases.

Precision or recall may be misleading when considered alone since they are interdependent. Thus, a combined measure is considered. The effective in terms of both precision and recall as follow:

$$F_{\propto} = \frac{1}{\propto . \frac{1}{precision} + (1-\propto) . \frac{1}{recall}}$$

where $0 \leq \propto \leq 1$. A value of a=0.5 is usually used, which attributes equal importance to precision and recall and is usually referred to as F1.

$$F1 = \frac{2 * recall * precision}{recall + precision}$$

The above definitions are applicable for each category. To obtain measures relating to all categories, micro-averaging and macro-averaging are used. In micro-averaging, the performance measures are obtained by globally summing over all individual decisions. In macro-averaging, the performance measures are first evaluated locally for each category, and then globally by averaging over the results of the different categories.

2.8 Summary

A thorough review to evaluate the breadth and depth of the issues pertaining to web page classification technology has been presented in this chapter. A typical web page classification process consists of steps such as feature selection, feature extraction, classifier design, and finally performance evaluation. Numerous feature selection methods, feature extraction methods, and classifiers have been proposed and were used for the web page classification problem. Empirical evaluations of classifiers such as Support Vector Machines (SVMs), k-Nearest Neighbor (k-NN), and Naïve Bayes (NB), have shown that these algorithms are effective in classifying small segments of web directories. However, previous web page categorization research on a few common categories or selected small segments of web taxonomies could not preserve the original characteristics of the web taxonomy as a whole. Hence, the observations from earlier studies do not take a broad view of this area.

The growing number of categories and web pages makes large-scale web page classification an inevitable component for web directories and search engines. Unfortunately, the few available web page classification researches on reasonably sized subsets of popular web directories conclude that in terms of effectiveness these classification algorithms cannot fulfill the classification needs of very large-scale taxonomies. Such web directories have hundreds of thousands of categories, deep hierarchies, class-imbalance and rarity over the hierarchies. The class imbalance and rarity make applying classification algorithms to such data sets very difficult and the problem has not been thoroughly studied.

While classifying very large datasets similar to web taxonomies, the impact of class imbalance and absolute rarity during the different stages of the classification process should be prevented or addressed. Various data level, algorithmic and architectural solutions for these issues has been proposed by the machine learning communities.

Chapter 3 of this thesis review critically analyzes the relative merits of these solutions in the context of large-scale web page classification.

CHAPTER 3: CLASSIFICATION OF VERY LARGE AND HIGHLY IMBALANCED DATASETS

3.1 Introduction

Traditional classification algorithms assume the target classes of the dataset share similar prior probability distributions. However, in real-world datasets like web taxonomies and intrusion datasets this identical prior probability assumption is violated (Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006; Monard, M. C., & Batista, G, 2002; Akbani, R., Kwek ,S., & Japkowicz, N., 2004; Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008; Japkowicz, N.,& Stephen, S, 2002; Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004; Zelikovitz, S., & Hirsh, H., 2000). This induces class imbalance within the dataset. Moreover, most of the categories in web directories have very few labeled documents of their own indicating rarity within the dataset. This leads to the situation of insufficient training instances to train the classifier. In the absence of a sufficient training set, the learning algorithm may find many different learning rules within the decision boundary all giving the same accuracy on the training data. In hierarchical datasets similar to web directories, expanding the content of each category using the web pages from the child categories helps to decrease the degree of rarity. This process, however, results in the localized overabundance of positive instances and the machine learning issues associated with large-sample learning. When classifying a very large dataset, it is critical to strike an optimum balance between the training dataset size and the associated solution complexity. Error estimates of a classifier can be improved by training a classifier on a very large dataset. This, on the other hand, will result in a large increase in solution complexity. The class imbalance, rarity and large-sample learning issues within the web directory make applying classification algorithms very difficult. This chapter reviews popular class imbalance, rarity handling and large-sample learning techniques. This includes sub-sampling, one-class learning, adaptive over-sampling, incremental sampling based learning and ensemble learning.

The overall goals of this chapter are to address the following queries:

1. What are the various data level and algorithmic solutions to address class imbalance?
2. What are the various data level and algorithmic solutions for classifying the rare categories?
3. What are the different complexity-effective techniques for learning from large dataset?
4. What are the main limitations of the earlier large scale web page classification research resulting poor classifier performance?

Section 3.2 of this chapter evaluates different data level and algorithmic solutions for the class imbalance problem. Various data level and algorithmic solutions for learning the absolute rare categories are discussed in Section 3.3. Different complexity-effective techniques for learning from large dataset are discussed in Section 3.4. In section 3.5, we provide a comprehensive review of the recent research activities in class imbalance and rarity. The limitations of earlier large-scale web page classification research are reviewed in Section 3.6. Section 3.7 is the summary of this chapter.

3.2 Machine Learning Issues While Classifying Highly Imbalanced Dataset

In contrast to the traditional benchmark datasets used for document classification, web directories, in general, have more categories and documents in the middle of the hierarchy than at either the upper or the lower levels of the hierarchy. This spindled category and document distribution is an indication of the class imbalance within the dataset.

The impact of the class imbalance problem varies across the domains. How class imbalance affects a particular task must be clearly understood in order to select an appropriate approach for the given task using the dataset. Considering the web page classification problem, an imbalanced dataset influences the different stages of the classification such as feature selection, classifier training and performance evaluation. When a classification algorithm is exposed to an imbalanced dataset, standard classifiers focus on the large classes and ignore the small classes. The effect of class imbalance on feature selection methods is not clear. Preliminary studies in this area conclude that implicitly combining positive and negative features using two-sided metrics like Information Gain is not inevitably optimal for imbalanced data. The Naïve Bayes classifier and Regularized Logistic Regression were used to set up these experiments. Classifiers trained on a well-judged combination of positive and negative features showed great potential and practical merits over the classifiers trained on the features identified by the Information Gain feature selection metric (Zheng, 2004).

The overall performance of a classifier and the appropriateness of the different classification technologies for a given dataset are analyzed with the help of the evaluation metrics. Accuracy is the most commonly used evaluation metric for classification tasks. Accuracy computes the fraction of examples that are correctly classified. In an imbalanced dataset, accuracy places more weight on the common classes than on rare classes. This makes the evaluation of the performance of a classifier on the rare classes difficult. An evaluation metric to analyze the performance of rare classes is needed. Different Studies by Weiss and Provost (Weiss, G. M., & Provost, F., 2003; Provost, F., & Fawcett, T., 2001) also highlight the necessity for an evaluation metric to evaluate the rare class performance of a classifier.

The most common evaluation metric used to evaluate the performance of a classifier trained on an imbalanced dataset is the Receiver Operator Characteristics (ROC). ROC is the relationship between True Positive Rate (TP Rate) and False Positive Rate (FP Rate). The area under the ROC curve, AUC is used to assess overall classification performance. A classifier with a high value of AUC is better. The ROC curve does not place more emphasis on one class over the other. So it is not biased against the minority class.

Other metrics from the Information Retrieval community that are useful to evaluate the performance of classifier trained on an imbalanced dataset are Recall and Precision. Precision is defined as the number of documents correctly assigned to a category divided by total number documents assigned to that category. Recall is defined as the number of correctly assigned documents into a category divided by total number of existing documents which should have been assigned. The value of Recall and Precision varies between zero and one. Compared to AUC, Precision and Recall have better solution interpretability. The Geometric mean and F1-Measure are the combined measures of recall and precision to give an overall performance measure. The Geometric mean is defined as the square root of precision times recall. The F1-Measure is the harmonic mean of recall and precision. Whereas accuracy ignores the performance of rare categories, ROC, Geometric mean and the F1-Measure do not. This is because both the TP Rate and the FP Rate are defined with respect to the positive rare class (Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006; Visa, S., & Ralescu, A., 2005; Japkowicz, N., & Stephen, S, 2002).

The evaluation metric defined with respect to TP Rate and FP Rate helps to evaluate the performance of the rare classes correctly. However, the impact of class imbalance during the different stages of the classification process should be prevented or treated. Various data level, algorithmic and architectural solutions for classifying an imbalanced dataset has been proposed. The next sections of this literature review critically analyze these solutions.

3.2.1 Changing Class Distribution

Data level solutions to class imbalance mostly involve changing the prior probability distribution of the dataset before applying the machine learning algorithm. These comprise different sampling methods. By altering the distribution of training examples, sampling attempts to eliminate or minimize the class imbalance. The basic sampling methods to address class imbalance include under-sampling and over-sampling. Under-sampling eliminates the majority-class examples. Over-sampling, in its basic form, duplicates the minority-class examples. Both these sampling techniques decrease the degree of class imbalance making the rare classes less rare. After under-sampling, classifier performance may degrade due to the potential loss of useful majority-class examples. Similarly, the additional training cases introduced by over-sampling can increase the time complexity of the classifier. In the worst case, exact copies of examples after over-sampling may lead to classifier over-fitting (Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006).

Advanced over-sampling methods, such as adaptive over-sampling and boosting, help to minimize the flaws of the basic sampling methods (Weiss, S. M., Apte, C., Damerau, F. J., Johnson, D. E., Oles, F. J., Goetz, T., & Hampp, T., 1999; Drummond, C., & Holte, R. C., 2003;). In adaptive sampling, new classifiers are iteratively induced by increasing the weight of erroneously classified cases. This is achieved by increasing the frequency of the wrongly classified cases in the next sample. Boosting, after every iteration, increases the weights associated with the incorrectly classified examples and decreases the weights associated with the correctly classified examples. In general, advanced over-sampling methods force the learner to concentrate more on the incorrectly classified examples in the next iteration. Any data level approach, however, results in the alteration of the prior probability distribution of the original dataset. A classification approach that preserves the original prior probability distribution of the dataset is always optimal. This includes different classifier level techniques such as manipulating classifiers internally and one-class learning.

3.2.2 Manipulating Classifiers Internally

This approach attempts to compensate for the imbalance in the training sample without changing the class distributions. Taking K-NN as an example, Barandela et al. (2003) proposed a weighted distance function to bias the discrimination procedure. In their research, weights are assigned to the respective classes and not to the individual training instances. This ensures a lower distance function to the positive minority class. A new case to be classified naturally tries to find their nearest neighbor among the learned instances of the minority class.

Bias to the discrimination procedure of the SVM is achieved mostly by moving the hyper-plane further away from the positive class. This compensates the skew associated with imbalanced datasets that pushes the hyper-plane closer to the positive class. This biasing is accomplished in different ways. Popular methods include changing the kernel function (Wu, G., & Chang, E., 2003), making errors on positive examples costlier (Veropoulos, K., Campbell, C., & Cristianini, N., 1999), and Biased Mini-max Probability Machine (BMPM) (Huang, K. Z., Yang, H. Q., King, I.,& Lyu, M. R., 2004).

Biasing the learned decision space is another popular approach for improving the classifier performance on class imbalance. In the basic form, this approach involves eliminating some small disjuncts from the learned decision space. A disjunct within a learned decision space can be visualized as an independent segment of the decision space formed by the association of a few decision rules. Thus, a small disjunct in a learned decision space is a relatively weak area of the decision space due to the minimal support from the training instances. The strength or significance of a disjunct is decided after a statistical significance test or application of error estimation techniques. Thus, only improperly learned disjuncts are removed from the decision space. Many papers discussed the interaction between the class imbalance and the small disjunct problems (Holte, R. C., Acker, L. E., & Porter, B. W., 1989; Weiss, G. M., & Hirsh, H., 2000). In certain cases, addressing the small disjunct problem without considering the class imbalance problem improves the classifier performance while for some other cases handling the small disjuncts alone is not sufficient to address the class imbalance problem.

3.2.3 One-Class Learning

Scholkopf et al. (2001) extended the Support Vector Machine algorithm to address training using only positive information known as "one-class" classification. In one-class SVM, the origin is treated as the member of the second class and the candidate class is separated from the origin. Thus, misconceptions on learning the negative dataset were alleviated. Raskutti et al. demonstrated the optimality of one-class SVMs over two-class ones by classifying the highly imbalanced genomic data (Raskutti, B. & Kowalczyk, A., 2004). They conclude that one-class learning is predominantly useful when used on extremely unbalanced datasets with a high dimensional noisy feature space. However, the performance of one-class SVM on very large datasets, such as web taxonomies, is not well studied.

3.2.4 Comparison of Different Class Imbalance Handling Techniques

Different data level and algorithmic solutions to address class imbalance problems were evaluated in this review. The critical question to be answered is "Which is the best approach?" There are no comprehensive empirical studies that evaluate all the aforementioned methods. Each research typically compares its method for handling class imbalance to a base classifier that has no special modifications for handling class imbalance. For extremely skewed datasets, under-sampling and over-sampling methods were often combined to improve generalization of the learner (Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006; G. Monard, M. C., & Batista, G, 2002). Batista et al. (2004) presented a comparison of various sampling strategies. They conclude that combining focused over-sampling and under-sampling is applicable when the data sets are highly imbalanced or there were very few examples of the minority class. Sampling techniques were extensively used in this area and there were a few studies (Drummond, C., & Holte, R. C., 2003; Japkowicz, N., & Stephen, S, 2002) that compare sampling methods. Unfortunately, even in this case the conclusions are not consistent.

3.3 Machine Learning Issues While Classifying Extremely Rare Categories

Most of the categories in web directories have very few labeled documents of their own. This indicates rarity within the dataset. This leads to the situation of insufficient training instances to train the classifier. In the absence of a sufficient training set, the learning algorithm may find many different learning rules within the decision boundary all giving the same accuracy on the training data. Data level or algorithmic treatments are necessary to learn the absolute rare classes of the web directory. Different approaches to address absolute rarity are discussed in the next section.

3.3.1 Changing Class Distribution

Over-sampling is a popular method to address absolute rarity. Over-sampling, in its basic form, duplicates the rare class examples. Thus, over-sampling techniques address absolute rarity by making the rare classes less rare. Exact copies of examples after over-sampling may lead to the classifier over-fitting. Moreover, basic over-sampling methods do not introduce new data. Hence, it does not address the fundamental lack of data issue associated with the minority class. This explains why some studies conclude that simple over-sampling is ineffective in improving minority classification (Drummond, C., & Holte, R. C., 2003; Ling, C.& Li, C., 1998).

Advanced over-sampling methods such as adaptive sampling (Weiss, S. M., Apte, C., Damerau, F. J., Johnson, D. E., Oles, F. J., Goetz, T., & Hampp, T., 1999) and Boosting (Zhou, 2008; Freund, Y., & Schapire, E., 1999) minimize the discussed flaws of the basic over-sampling. If the rare classes are more error-prone than common classes, it is quite logical to believe that boosting may improve classification performance of the rare classes. Boosting, however, is not well studied in the context of rarity. The advanced sampling methods make boosting and adaptive re-sampling free from information loss and classifier over-fitting issues. Hence, these methods may outperform the traditional random over-sampling and random under-sampling techniques.

3.3.2 Non-Greedy Search Techniques

Classifiers like the Decision Tree employ a divide-and-conquer approach, where the original problem is decomposed into smaller and smaller problems. Thus the instance space is partitioned into smaller and smaller pieces after every iteration. This process leads to data fragmentation (Friedman, J. H., Kohavi, R., & Yun, Y., 1996). Due to the existing lack of data, data fragmentation is an extreme concern on learning from rare classes (Weiss G. M., 2004.). Non-greedy search methods were also explored to address absolute rarity. A popular non-greedy method involves genetic algorithms. Genetic algorithms are global search techniques that work with populations of candidate solutions rather than a single solution and employ stochastic operators to guide the search process (Weiss G. M., 2004.). These characteristics permit genetic algorithms to deal well with attribute interactions to avoid being stuck in local maxima. Hence genetic algorithms may be suitable for dealing with absolute rarity.

3.3.3 Ensemble Learning

The functioning of a learning algorithm can be visualized as a search process on a decision space to identify the best learning rule. The problems arise when the amount of training data is too small compared to the decision boundary. In the absence of sufficient data, the learning algorithm may find many different learning rules within the decision boundary all giving the same accuracy on the training data. An appropriate architecture may help to reduce the risk of choosing an inappropriate decision rule from the learned hypothetical space. By constructing an ensemble of multiple member classifiers with different feature distributions, the algorithm can average the results to reduce the risk of choosing a very wrong single classifier (Dietterich, 2000).

3.4 Machine Learning Issues While Classifying a Dataset with Very Large Number of Positive Training Instances

In contrast to the traditional benchmark datasets, web directories generally have a very large number of positive and negative instances, especially in the upper-level hierarchies. Hence, the machine-learning algorithms have to extract knowledge from very large datasets. In theory, most prediction methods should perform well with a large dataset due to the increased enumeration and search space. Practical limitations like dimensional and computational issues, however, limit the advantage gained by using large datasets. In addition, more exhaustive searches may also increase the risk of finding a single low probability solution that evaluates well, but may fail to classify a new web page. Different data level and architectural approaches has been proposed to address the abundance of the training instances. This includes sub-sampling, incremental sampling based learning and ensemble learning

3.4.1 Sub-Sampling

Sub-sampling is the process of drawing a subset of data from a large dataset. Commonly used sampling methods include random sampling and stratified sampling. Numerous sub-sampling techniques from probability theory and statistics have been studied and their relative merits on classification accuracy and solution complexity have been experimentally evaluated. Catlett, (1991) investigated the effect of sub-sampling using random sampling and stratified sampling principles. In this work, learning from the subsets showed a decrease in classification accuracy and processing time. This observation is drawn from datasets of less than 100,000 records and thus cannot be generalized.

The relationship between training data size and prediction accuracy of the classifier is further investigated by Harris-Jones, C., & Haines, T.L. (1997). They evaluated two large business data sets of 300,000 records using the decision-tree learner C4.5 and its successor C5 and an increase in accuracy across the entire range of the dataset was found. The improvement in accuracy at the upper size limit, however, is quite small. The benefit of such a small improvement is questionable given the associated solution complexity due to increased search space and processing time. The major flaw with the above-mentioned sub-sampling approaches is the loss of information caused by ignoring a portion of the original dataset. The impact of information loss due to sub-sampling is minimized in Ensemble learning and Incremental sampling based learning (Weiss, S. M., & Indurkhya, N., 1998.), two accepted architecture level solutions for learning large datasets.

3.4.2 Incremental Sampling and Learning

The incremental sampling based learning (Weiss, S. M., & Indurkhya, N., 1998) is a multi-phase process. This includes training a single classifier on an increasingly larger random subset of cases, observing the trends and stopping when no progress has been made. The subset should take big bites from the original data to ensure the chances of improving performance with more data. The central theme is to observe the trends and net change in error. A decision on whether further experiment is necessary is made prior to the next increment of the dataset size. A significant amount of new data on every iteration should lead to better performance along with an acceptable system complexity.

Taking an example, in the first step two classifiers are designed taking 10% and 20% of the cases from the original dataset. The trend of the classifier, in terms of error and solution complexity on moving from 10% cases to 20% cases is analyzed using test cases. If the error decreases with increased dataset size or if the current complexity is acceptable to the maximum of the expected solution complexity, the subset size is further increased to 33% of the records. If necessary, the sample size can be increased gradually to the full sample of cases. For a given sub-set size, if the error has not decreased, increasing the subset size will not be effective. To ensure a stabilized classifier, the trend of the error can be noted on increasing the numbers of samples until a plateau is reached.

3.4.3 Ensemble based Average Sampling

In ensemble learning, sub-sampling is performed on the original dataset to create multiple training datasets. Instances of each training dataset are varied from that of the original training dataset. Member classifiers are developed from each training dataset. An ensemble classifier is thus comprised of multiple member classifiers. The classification result by the ensemble classifier is determined by voting by the member classifiers. The voted solution typically has less error than a single solution. On a sufficient sample size, the average of their solution can produce the same result as much larger samples (Weiss, S. M., & Indurkhya, N., 1998). In addition to reducing the average error by voting, ensembles on a large-scale learning process has the added advantage of increased accuracy compared to the individual classifiers. In a classification process, it is very difficult for a machine learning algorithm to find the best hypothesis on a very large dataset. On searching for the best hypothesis, many learning algorithms such as neural network and decision tree may stick in local optima. An ensemble constructed by running the algorithm on different segments of the data set may provide a better approximation to the true unknown test data compared to any of the individual classifiers (Dietterich, 2000).

3.4.4 Comparison of Different Large-Sample Learning Solutions

Irrespective of the consequences, sub-sampling of the original dataset is essential for learning from a very large dataset. Therefore, it is important to identify the most efficient sub-sizing treatment for a given dataset. Producing random samples efficiently from large data sets is a difficult process. Considering a very large dataset, if a sampling methodology has to scan the complete dataset in order to randomly sample, much of the benefit of sampling will be lost. Despite numerous investigations on the effect of different sub-sampling approaches on classification accuracy and processing time, no consensus has been reached on which approach is optimal for a given classifier or for a given range of dataset size. From the classification perspective, the sub-sampling method without disturbing the inherent prior probability distributions of the original dataset may be more appropriate.

3.5 Learning Imbalanced and Rare Dataset: Progress and Prospects

In this section, we provide a comprehensive review of the recent research activities in the areas of class imbalance and rarity.

In general, sampling methods are effective data level solution to address class imbalance and rarity (Seiffert, C., Khoshgoftaar,T.M., Van Hulse,J., & Napolitano,A., 2007). The basic sampling methods include under-sampling and over-sampling. As mentioned earlier under-sampling randomly discards the majority class samples while over-sampling randomly duplicates the minority class samples in order to modify the class distributions. Although these two methods do alleviate the rarity and class imbalance problem to some extent, they have some limitations too. In random under-sampling, some potentially useful majority samples may be left out, resulting in information loss and a less than optimal model. Also, in random over-sampling, the size of the training set may increase significantly, increasing the computational complexity. Moreover, the exact duplicate instances after over-sampling may cause the over-fitting problem (Mease, D., Wyner,A.J., & Buja,A., 2007).

Recently, different algorithmic and architectural solutions to alleviate the issues associated with sub-sampling when applied to address class imbalance and rarity have proposed and used by the data mining and machine learning community. Most of these solutions either try to minimize the negative effects of sampling or address the class imbalance and rarity using appropriate machine learning paradigms.

For example, Liu et al, (Liu, X.Y., Wu,J., & Zhou.Z.H., 2009) proposed two under-sampling methods, EasyEnsemble and Balance Cascade, to address the information loss introduced in the traditional random under sampling method. They created multiple subsets from the majority class, combined with rare class dataset and used AdaBoost to train classifiers for each subset. Finally, the outputs of these classifiers are combined. Experimental results suggest that, compared to EasyEnsemble, BalanceCascade is more efficient on highly skewed dataset.

Chawla et al, (Chawla, N.V.,Bowyer,K.W.,Hall,L.O., and W. P. Kegelmeyer., 2002) proposed an approach called Synthetic Minority Over-sampling Technique (SMOTE). In this research, the rare classes are over-sampled using synthetic rare class samples. The synthetic rare class samples are generated by applying k- nearest neighbour approach on the rare samples. This prevents classifier over-fitting due to the exact duplicates of the rare instances. Experimental results conclude that SMOTE can improve the accuracy of classifiers on many rare class problems. Also, the combination of SMOTE and under-sampling performed better than pure under-sampling.

ADASYN, another recent research in this area, make use of density distribution function to adaptively create different amounts of synthetic data (He, H., Bai,Y.,Garcia,E.A., & Li,S., 2008). The density distribution function determines the number of synthetic samples that need to be generated for each minority example by adaptively changing the weights of different minority examples to compensate for the skewed distributions.

It is evident that synthetic sampling methods are quite strong in dealing with learning from imbalanced dataset. However, the synthetic data generation methods are mostly complex and computationally expensive. Mease et al, (Mease, D., Wyner,A.J., & Buja,A., 2007) proposed a much simpler technique for creating synthetic data. Instead of generating new data from computational methods, they applied “perturbations” on the duplicate data obtained from random sampling. This approach breaks the issue of exact duplicates with minimum computational cost. This idea is relatively simple compared to its synthetic sampling counterparts and also incorporates the benefits of boosted ensembles to improve performance.

The empirical studies suggest that synthetic procedures are successful in dealing with learning from imbalanced dataset if it does not jeopardize the runtime costs. And also, intelligently sub-sampled systems like EasyEnsemble, BalanceCascade and SMOTE claims some advancements compared to their counterpart systems with basic sub-sampling. However, while conducting sampling, how to determine the proper sampling rate, which directly affects the class distribution ratio, is not known. Another popular approach, cost-sensitive learning, is meant to address these issues by integrating cost information during the machine learning process.

Cost-sensitive learning is a widely used technique in data mining, where different levels of misclassification penalty are assigned to each class. Experimental studies have shown that in some application domains cost-sensitive learning is superior to the sampling methods (Liu, X.,Y., and Zhou,Z.,H., 2006), (McCarthy, K.,Zabar,B., & Weiss,G.M., 2005), (Liu., X.Y & Zhou.,Z.H, 2006). The cost-sensitive techniques, when integrated into the classification algorithms attempts to optimize the overall cost of misclassification using the cost information provided. Recently, this technique has been applied to the rare class problem in which a higher cost is given to the misclassification of rare objects compared to the majority class. For example, Statistical Online Cost Sensitive Classification (STOCS) (Zhao, J.H., Li,X., & Dong,Z.Y., 2007) is proposed to classify rare events as online. Hence, cost-sensitive learning techniques can be considered as a viable alternative to sampling methods for imbalanced learning domains.

Many research works apply general sampling and ensemble techniques to the SVM framework. Some examples include the SMOTE with Different Costs (SDC) (Akbani, R.,Kwek,S., & Japkowicz,N., 2004) and the ensembles of over or under sampled SVMs (Vilarino, F., Spyridonos,P., Radeva,P., & Vitria,J., 2005), (Kang, P. & Cho,S., 2006), (Liu, Y., An,A., & Huang,X., 2006), (Wang, B.,X., & Japkowicz,N, 2008), (Tang, Y., & Zhang,Y.,Q., 2006).

The SDC algorithm uses different error costs (Akbani, R.,Kwek,S., & Japkowicz,N., 2004) for different classes. Such dataset when trained on SVM biases the algorithm and imposes an additional shift in the decision boundary farther from positive instances. This approach makes positive instances more densely distributed and there by ensure a more well-defined boundary. The methods proposed in (Kang, 2006) and (Liu Y. A., 2006) develop ensemble systems by modifying the data distributions without modifying the underlying SVM classifier.

Empirical studies conclude that cost sensitive learning is effective to address class imbalance and rarity issues associated with the data classification. However, in practice, it is often difficult to set the cost information. It is well known that a false negative prediction is more risky than a false positive prediction. However, to make a quantitative analysis between these two risk factors require prior knowledge on the domain or domain experts' involvement.

Due to the aforementioned underlying issues associated with the sampling and cost sensitive learning, different algorithmic solutions to address class imbalance and rarity also have been investigated.

Wang and Japkowicz (Wang, B.,X., & Japkowicz,N, 2008) proposed a modified SVM architecture with asymmetric misclassification costs to address class imbalance. This approaches use an iterative procedure to effectively modify the weights of the training observations. In another research, Japkowicz (Japkowicz, N., 2002) attempted to formulate rare class less rare through sub-division. Author claims that after sub-division the complex concept become much simpler and this result into the reduction in the degree of imbalance. This research used both unsupervised and supervised learning in classification tasks where division is implemented via clustering.

Wu et al., (Wu, J., Xiong, H., Wu, P., & Chen, J., 2007) also adopted the idea of sub-division by developing a method for rare class mining using local clustering. For the majority classes, local clustering is employed within each class, and for the rare class, over-sampling is adopted. The algorithm adjusts the over-sampling parameter to fit in with the clustering result so that the rare class size is approximate to the average size of the partitioned majority class.

The effectiveness of Active Learning methods to address class imbalance also have been investigated. Ertekin et al. (Ertekin, S., Huang,J., Bottou,L., & Giles, L., 2007) and (Ertekin, S., Huang, J., & Giles, C.L., 2007) proposed an efficient SVM-based active learning method. This is a multi-phase procedure which include training an SVM using the complete dataset, identification of the most informative instances based on the hyperplane distance and training another SVM using the a new training set formed by the most informative instances . In general, for each instance of the dataset, the algorithm needs to recalculate the distance from the current hyper-plane, makes this approach computationally expensive. To solve this problem, they proposed a method to effectively select such informative instances from a random set of training populations. Additionally, early stopping criteria for active learning are also discussed in this work which can be used to achieve faster convergence of the active learning process as compared to the random sample selection solution.

Another active learning sampling method is the Simple Active Learning Heuristic approach proposed by Doucette et.al. The key idea of this method is to provide a generic model for the evolution of genetic programming classifiers by integrating the stochastic sub-sampling method and a modified Wilcoxon-Mann-Whitney cost function (Doucette, J., & Heywood,M.I., 2008).

Solutions to handle the imbalanced learning problem are not exclusively in the form of sampling methods, cost-sensitive methods, kernel-based methods, and active learning methods. For instance, the one-class learning or novelty detection methods have also attracted much attention in the community.

Generally speaking, this group of approaches aims to recognize instances of a concept by using a single class of examples rather than differentiating between instances of both positive and negative classes as in the conventional learning approaches. Recent representative works in this area include the one-class SVMs (Zhuang, L., & Dai,H., 2006), (Lee, H.,J., & Cho,S., 2006), (Zhuang, L., & Dai, H., 2006) and the autoassociator (or autoencoder) method (Manevitz, L., & Yousef, M., 2007). Lee and Cho (Lee, H.,J., & Cho,S., 2006) proved that novelty detection methods are particularly useful for extremely imbalanced data sets.

The Mahalanobis-Taguchi System (MTS) has also been used for imbalanced learning (Su, C.T., & Hsiao,Y.H., 2007). The MTS was originally developed as a diagnostic and forecasting technique for multivariate data. Learning in the MTS is performed by developing a continuous measurement scale using single class examples instead of the entire training data. Hence, MTS model will be least influenced by the skewed data distribution. Due to the same reason, MTS may provide robust classification performance when applied to the skewed dataset classification. Su et.al compared the effectiveness of the MTS model for imbalanced learning with Stepwise Discriminate Analysis (SDA), Backpropagation Neural Networks, Decision Trees, and SVMs. This work demonstrates MTS as an effective solution to imbalance dataset classification.

Most of the research works so far we examined addresses either class imbalance or small sample size problem (rarity). In many of today's data analysis and knowledge discovery applications, it is often unavoidable to have data with high dimensionality, class imbalance and small sample size. Some specific examples include web directory classification; face recognition and gene expression data analysis, among others. In this situation, there are many critical issues that arise simultaneously.

If the sample size is small, all of the issues related to lack of sufficient training instances and class imbalances are applicable. The combination of class imbalance, rarity and high dimensionality hinders the classifier performance. This is because of the difficultly involved in forming conjunctions over the high degree of features with limited samples and class imbalance. This issue requires more attention in the machine learning and data mining community.

Finally, while this review focused on two-class imbalanced problems, multiclass imbalanced learning problems exist and are of equal importance. Sun et.al, proposed a cost-sensitive boosting algorithm AdaC2.M1 to tackle the class imbalance problem with multiple classes (Sun, Y., Kamel, M.S., & Wang,Y., 2006). They used genetic algorithm to fix the the optimum cost setup for each class. Abe et.al, proposed an iterative method for multiclass cost-sensitive learning. The key idea of this approach includes iterative cost weighting, data space expansion, and gradient boosting with stochastic ensembles (Abe, N., Zadrozny, B., & Langford,J., 2004). Chen et.al, proposed a min-max modular network to decompose a multiclass imbalanced learning problem into a series of small two-class sub-problems (Chen, K., Lu, B.L., & Kwok, J., 2006).Other works of multiclass imbalanced learning include the rescaling approach for multiclass cost-sensitive neural networks (Zhou, Z.,H., & Liu, X.Y., 2006), (Liu, X.Y. & Zhou, Z.H., 2006) and others.

3.6 Limitations of Earlier Large Scale Web Page Classification Research

As of our knowledge, the maximum number of web page categories ever classified is not more than 246,279 categories from Yahoo! (Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004). In their research, even with the best setting, hierarchical SVMs lead to a Micro-F1 of 24%. However, the hierarchical classifier evaluation procedure they followed to calculate the reported Macro-F1 measure is not clear. This section investigates the reasons for the poor performance reported by earlier large-scale web page classification research.

To start with, we conducted a statistical analysis of Yahoo! and DMOZ web directories. These web directories have hundreds of thousands of categories, deep hierarchies, class imbalance and rarity (very small classes) within the dataset. These properties make applying classification algorithms to such data sets very difficult.

For example, in the Yahoo! web directory, 0.16% of the total categories (988 categories) are very large, containing 1000 to 600,000 labeled web pages of their own; whereas, 19.06% of the Yahoo! categories are absolutely rare categories of 10 to 100 labeled web pages. Classification algorithms, when applied to very large categories of more than 1000 labeled instances should address the machine learning issues due to the class imbalance and large-sample learning. Conversely, classification algorithms, when applied to rare categories of 10 to 100 labeled web pages should address the machine learning issues due to class imbalance and rarity. Another 1.58% of Yahoo! categories are reasonably sized categories holding 100 to 1000 labeled web pages. However, the abundance or shortage of negative instances in the sibling categories makes these categories imbalanced. There are 504,240 Yahoo! categories containing 1 to 9 web pages. This forms 78.82% of the total Yahoo! categories.

Earlier large scale web page classification researches either overlooked the machine learning issues due to rarity, class imbalance and over-abundance of training instances or addressed these issues using a common methodology. These researches either lead to highly comprehensive or inadequate statements such as traditional web page classification techniques are insufficient to address the challenges of large-scale web page classification problem, or provide limited insight to the real challenges of large-scale web page classification.

3.7 Summary

A thorough review to evaluate the breadth and depth of the issues pertaining to large-scale web page classification technology has been presented in this chapter. These include localized over-abundance of positive training instances, class imbalance and rarity. Different data level, algorithmic, and architectural solutions to these issues have been proposed and were used by the machine learning community. This chapter analyzes the effectiveness of these approaches in the context of large-scale web page classification.

CHAPTER 4: METHODOLOGY

4.1 Introduction

Web directories have hundreds of thousands of categories, deep hierarchies, class imbalance and rarity within the dataset. These properties make applying classification algorithms to such data sets very difficult. In this chapter, we investigate a scalable and effective classification methodology for large-scale web page classification.

Web page processing is the first step of any content based web page classification process. Web page preprocessing for a content-based classification integrates various methods to identify the concise portion of the web pages with maximum textual information, applicable to the majority of the web pages. Different web page preprocessing approaches using the HTML structure and hypertext structure of the web pages have been studied and their effectiveness in the context of web page classification has been evaluated. However, drawing general conclusions in web page preprocessing can be misleading (Yang. Y, 2001). Hence we examined the quality and nature of textual information across the various HTML tags prior to the web page preprocessing. The distributed version of the popular MapReduce algorithm, Apache Hadoop, is used for this purpose. The distribution of the textual information across the different tags of the Yahoo! Web pages is summarized as Table 2.

Table 2: Feature Distribution within the HTML Tags of Yahoo! Web Directory

Tag type	**0 words (%)**	**1-5 words (%)**	**5-50 words (%)**	**51+ words (%)**
Title	62.5	27.30	9.001	1.20
Meta Description	64.9	16.85	15.21	3.04
Meta Key-word	60.07	14.02	12.21	13.7
Body text	22.1	17.5	18	52.4

In the Yahoo! Web directory, the most obvious source of textual information for the purpose of classification is the body of the web pages. However, 22.1% of the web pages have no usable body text. About 52.4% of web pages contain more than 50 words within the body of the web page, and 25.5% of the web pages contain 1 to 50 words within the body of the web page. Other sources of text are the content in HTML tags including titles, Meta-key word and Meta-Description. However, the amount of text within these tags is relatively very small. Hence we used the textual content from the body of the web pages for the purpose of classification. The following steps have been executed to remove the less informative contents of the textual information within the body segment.

1. Removing HTML tags and scripting languages such as java script
2. Removing stop words
3. Word stemming

The textual information remaining after these steps is known as features. Using the features and their frequency of occurrences, the web pages of each category is projected into a multidimensional space.

4.2 Statistical Analysis of Yahoo! Web Directory

In contrast to the traditional benchmark datasets used for document classification, the Yahoo! web directory has more categories and documents in the middle of the hierarchy than at either the upper or the lower levels of the hierarchy. This spindled category and document distribution is an indication of the class imbalance within the dataset. In an imbalanced dataset, almost all examples belong to one class, while far fewer examples represent the other classes.

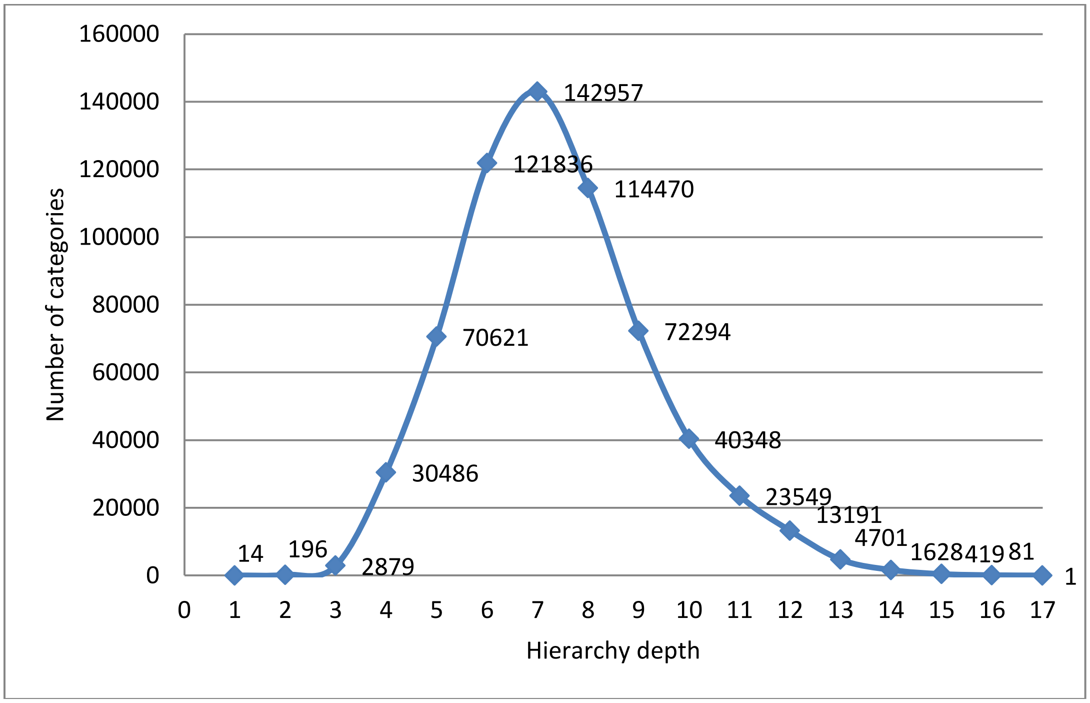

Figure 1: Spindled Category Distribution of Yahoo! Web Directory

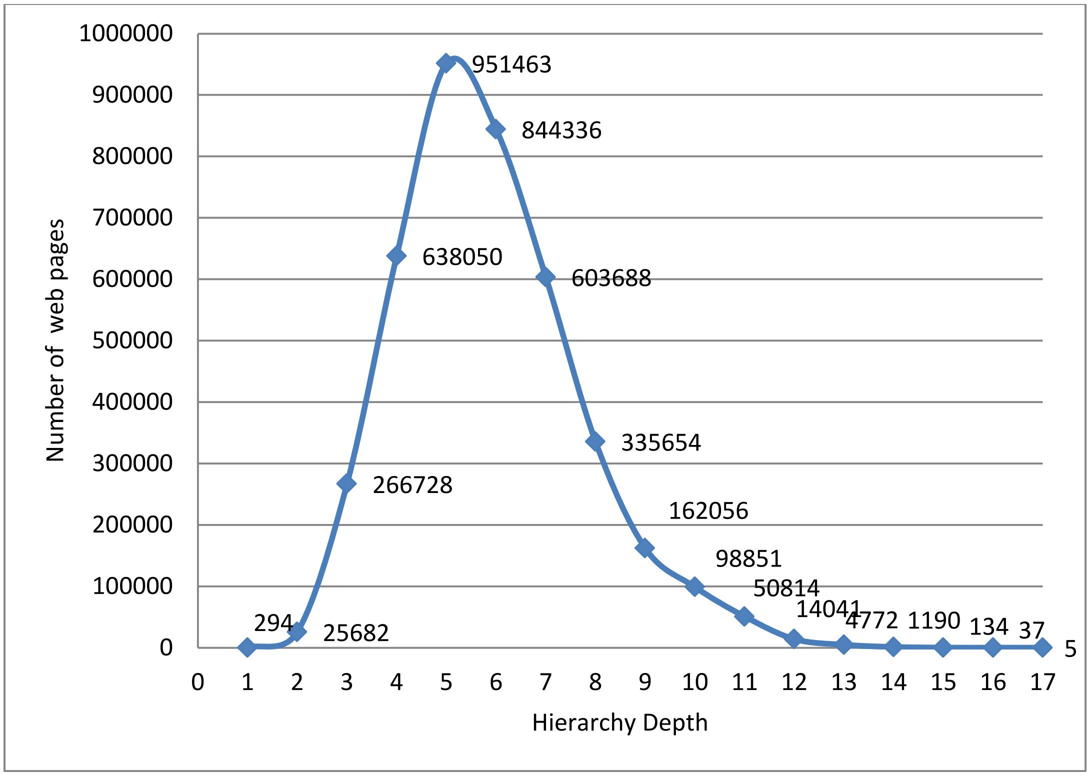

Figure 2: Spindled Web Page Distribution of Yahoo! Web Directory

Class imbalance within the dataset affects different stages of the classification process including feature selection, classifier training and performance evaluation. The impact of class imbalance on popular feature selection methods is beyond the scope of this research. However, preliminary research in this area is conducted to compare the optimality of the two-sided metric Information Gain with the one-sided metric Document frequency.

When the classification algorithm is exposed to an imbalanced dataset, standard classifiers tend to focus on the large classes and ignore the small classes. In addition, popular evaluation measures such as accuracy place more weight on the common classes than on rare classes. Thus, the performance with respect to small classes is difficult to assess.

Various data level and algorithmic solutions for classifying imbalanced datasets have been proposed and used by the machine learning community. This includes different sub-sampling techniques and one-class learning. However, there are no comprehensive empirical studies that evaluate these methods.

We addressed the class imbalance problem by focused over-sampling and under-sampling of the negative instances. This prevents information loss due to the sub-sampling of positive instances. One-class learning is another popular algorithmic approach that we examined to address class imbalance. In one-class learning, the origin is treated as the member of the second class and the candidate class is separated from the origin. Hence, the misconceptions on learning the negative dataset will be alleviated.

The macro-averaged F1-measure is used to evaluate the performance of the rare classes correctly. Whereas accuracy, the popular evaluation metric for classification, ignores the performance of rare categories, the F1-Measure, the harmonic mean of recall and precision, does not. This is because both the TP Rate and the FP Rate used to calculate recall and precision are defined with respect to the positive class.

Another distinguishing attribute of the Yahoo! web directory is the skewed category distribution over the web pages. The skewed category distribution on the number of web pages indicates that most categories have very few labeled web pages. The lack of sufficient training instances for classification indicates rarity within the dataset. Data level or algorithmic approaches prior to or along with machine learning is necessary to learn the rare categories of the web directory. A popular solution to address absolute rarity is over-sampling. However, exact copies of examples after basic over-sampling may lead to classifier over-fitting. Advanced over-sampling methods such as adaptive over-sampling minimize the flaw of the basic over-sampling. Hence, these methods may outperform the traditional random over-sampling. The effectiveness of adaptive over-sampling to address rarity is examined in this research.

To decrease the overall rarity and the number of conceptual nodes, the content of each Yahoo! category is expanded using the web pages from the child categories. This is performed prior to the feature space reduction and classification. This process, however, results in the localized overabundance of positive instances especially in the upper level categories of the Yahoo! web directory.

Machine learning algorithms when applied to very large datasets have to extract knowledge from a very large decision space. In theory, most prediction methods should perform well with a large dataset due to the increased enumeration and search space. Practical limitations like dimensional and computational issues, however, limit the advantage gained by using large datasets. In addition, more exhaustive searches may also increase the risk of finding a single low probability solution that evaluates well, but may fail to classify a new web page. Different data level and architectural solutions including sub-sampling, incremental sampling based learning and ensemble learning have been proposed and used to address these issues. The major flaw with sub-sampling approaches is the information loss caused by ignoring a portion of the original dataset. The effectiveness of Incremental sampling based learning and Ensemble learning when applied to very large categories with hundreds of thousands of positive instances is compared in this research.

Another distinguishing characteristic of the Yahoo! web directory is web pages with multiple labels. The highest number of labels assigned to a Yahoo! web page is 35 and the average number of labels assigned to a web page within the Yahoo! web directory is 3.2. The existence of multiple labels points to the necessity of a classification frame-work that assigns multiple labels to the web pages.

4.3 Architecture

In an imbalanced dataset like the Yahoo! web directory, the prior probability distribution of a category indicates the presence or absence of class imbalance, rarity and large-sample learning issues due to the overabundance of positive instances. Based on the prior probability distribution and associated machine learning issues, we subdivided the entire Yahoo! subcategories into 5 mutually exclusive groups as given in Table 3.

Table 3: The Prior Probability Distribution of Web Pages within Yahoo! Web Directory

Group	Number of categories	% of categories
Categories with more than 1000 labeled web pages	988	0.16
Categories with 100 to 1000 labeled web pages	10,025	1.58
Categories with 1 to 10 web pages	504,240	78.82
Categories without any labeled web pages	3,089	0.38

The rationale for these 5 class sizes are explained fully in chapter 8. Later, different data level, algorithmic and architectural solutions are applied to address the machine learning issues associated with these 5 groups of categories. The best performing classification technologies for each group of categories following a particular prior probability distribution has been identified and applied for Yahoo! Web directory classification. The different machine learning issues associated with the Yahoo! Web directory is summarized as Table 4.

In the Yahoo! web directory, 0.16% of the total categories (988 categories) are very large, containing 1000 to 600,000 labeled web pages of their own; whereas, 19.06% of the Yahoo! categories are absolutely rare categories of 10 to 100 labeled web pages. Classification algorithms, when applied to very large categories of more than 1000 labeled instances should address the machine learning issues due to the class imbalance and large-sample learning. Conversely, classification algorithms, when applied to rare categories of 10 to 100 labeled web pages should address the machine learning issues due to class imbalance and rarity. Another 1.58% of Yahoo! categories are reasonably sized categories holding 100 to 1000 labeled web pages. However, the abundance or shortage of negative instances in the sibling categories makes these categories imbalanced. There are 504,240 Yahoo! categories containing 1 to 9 web pages. This forms 78.82% of total Yahoo! categories. Due to the extreme lack of training instances, no individual classifiers are designed for this group.

Table 4: Machine Learning Issues Associated with Yahoo! web Directory and Experimented Solutions

Machine Learning Issues	Investigated solutions
Categories with over-abundance of positive training instances together with negative dominant class imbalance	1. Ensemble learning architecture combined with sub-sampling of negative instances. 2. Incremental sampling and learning combined with sub-sampling of negative instances.
Categories with negative/positive dominant class imbalance	Single classifier with three-fold cross validation using complete positive instances and under/over sampled negative instances.
Rarity together with negative/positive dominant class imbalance	1. Adaptive over-sampling. 2. Random over-sampling
Extreme rarity	No separate classifier is designed. Web pages of these categories are evaluated by the classifiers of the parent categories

This research is conducted on Atlantic Canada Excellence Network (ACENet) High Performance Computing (HPC) environment. The tailoring, integration and distribution of different machine learning architectures is performed using java, FJTask Framework, Apache Hadoop, Apache Ant and Matlab Distributed Computing Toolbox.

We used a top down approach, starting at the root of Yahoo! web directory and a category includes the subcategories of that category. To start with, preprocessing of the Yahoo! category web pages is conducted using 165 clusters of ACENet parallel computing environment. The distributed version of MapReduce Algorithm, Apache Hadoop is used for this purpose.

Later, for each Yahoo! subcategory, say C_i, the collective positive instances by adding the positive instances of the C_i and all child category web pages is calculated. If the collective positive instance of C_i is greater than or equal to 1000, the category C_i is considered as large category and machine learning architectures described in Section 4.4 is applied on C_i. If the collective positive instance of C_i is between 100 and 1000, C_i is considered as reasonable sized but imbalanced category and the machine learning architecture mentioned in Section 4.5 is applied on C_i. Similarly if the collective positive instance of C_i is between 10 and 100, C_i is considered as rare category and the machine learning architecture mentioned in Section 4.6 is applied. The machine learning progresses in a layered mode starting from the first layer of 14 categories.

The Incremental Sampling Based Learning and Ensemble Learning architecture designed to address the over-abundance of positive instances associated with the Yahoo! web directory categories of more than 1000 positive instances are discussed in Section 4.4. The focused over-sampling and under-sampling procedure applied to address class imbalance associated with the Yahoo! categories of 100 to 1000 labeled instances is discussed in Section 4.5. The adaptive over-sampling architecture designed to address rarity within the Yahoo! web directory is discussed in Section 4.6.

4.4 Classification of Very Large Yahoo! Categories with more than 1000 Web Pages

4.4.1 Ensemble Learning Architecture

In ensemble learning, sub-sampling is performed on the original dataset to create multiple samples. Member classifiers are trained from each sample. Thus, an ensemble classifier comprises a group of member classifiers. The category of a new web page is determined by voting by the member classifiers. If sufficient training instances have been taken, the ensemble formed by the multiple member classifiers can produce the same result as much larger samples (Weiss, S. M., & Indurkhya, N., 1998). The optimal sample size varies with the dataset. The sample size optimization followed in this research includes training multiple member classifiers taking increasingly larger samples, observing the trends, and stopping when no progress has been made. The sample size used in the first iteration is ensured to be the representative of the original dataset. The size of the samples is increased gradually in the coming iterations to ensure the chances of improving performance with larger sample size.

While classifying a very large category, say C_i, of collective positive instances more than 100,000, in the 1st iteration, 2% of the labeled instances from C_i and all child categories of C_i are drawn and combined with an equal number of negative instances from the sibling categories of C_i. Thus, 50 unique samples of C_i are generated for member classifier design. To make stratified sampling of extremely rare categories with 1 to 10 web pages easier, all child categories of C_i of 1 to 5 web pages and 5 to 10 web pages are combined separately and treated as two single rare categories of category C_i.

In the 2nd iteration, 4% of the labeled instances from C_i and all child categories are drawn and combined with an equal number of negative instances from the sibling categories of C_i and 25 unique samples of C_i are generated for member classifier design.

In the 3rd, 4th, 5th, 6th, 7th, and 8th iterations 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories are drawn and combined with an equal numbers of negative instances from the sibling categories of C_i. In these iterations, 18, 12, 10, 5, 3 and 2 unique samples of C_i are created for member classifier design.

While classifying categories of 10,000 to 100,000 collective positive instances, in the 1st, 2nd, 3rd, 4th, 5th, and 6th iterations, 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories of C_i are drawn and combined with an equal number of negative instances from the sibling categories of C_i resulting in 18, 12, 10, 5, 3 and 2 unique samples of C_i for member classifier design.

For categories of 1000 to 10,000 positive instances, in the 1st, 2nd, 3rd, and 4th iterations, 10%, 20%, 33.33% and 50% of the positive instances from C_i and all child categories are drawn and combined with equal numbers of negative instances from the sibling categories of C_i resulting in 10, 5, 3, and 2 unique samples of C_i for member classifier design.

In this procedure, it is important to analyze the achieved benefit before moving to the next iteration of a higher sample size. For each member classifier, the average percentage of True Positive (TP) Rate and False Positive (FP) Rate after 3-fold cross validation is calculated. A quality factor, defined as a function of the average TP Rate and FP Rate of the member classifiers and the variance of the TP Rate and the FP Rate is used to measure the quality of ensembles formed on every iteration. A high TP Rate and low FP Rate is desirable for giving a classifier with high recall and precision. The higher the variance, the greater will be the dissimilarity between the member classifiers. Hence the Quality Factor (QF) of an ensemble formed of N member classifiers is defined as,

QF=Average of percentage TP Rate / (Average of percentage FP Rate × Average variance across the TP Rate × Average variance across FP Rate).

The slope in degrees between two consecutive QF's (y-axis) and the normalized average sample size (x-axis) is measured. A slope of zero degrees means there is no improvement in the classifier performance between the two consecutive sample sizes and no further iteration is needed. However, this is an optimal situation that rarely happens. For each sample, three sets of experiments have been conducted after fixing predefined thresholds to the slope as 1 degree, 2 degrees and 5 degrees. If the slope in degrees between two consecutive QF's and the average sample size is less than or equal to a predefined threshold say x degrees, the member classifiers formed from the higher sample size is taken as the optimal ensemble classifier for C_i. These member classifiers will be used further for the voting purpose to determine the category of a new web page. Otherwise the algorithm proceeds to the next iteration.

In a large scale classification application with hundreds of categories, occasionally the ensembles may fail to converge for the predefined slope cut-off. In such cases, the ensemble formed by the middle iteration of three consecutive iterations is taken as the optimal ensemble for classification, provided the difference of the two consecutive slopes calculated from these three consecutive iterations is less than or equal to 1 degree. Otherwise a single classifier is trained for the category taking complete positive instances and an equal number of negative instances are trained.

The effectiveness of this ensemble architecture, combined with Information Gain and Document Frequency feature selection methods and the Perceptron, Support Vector Machine and Maximum Entropy Classifiers have been studied and the best performing classification technologies have been applied to classify the Yahoo! web directories of more than 1000 positive instances.

Of the total 988 very large Yahoo! categories, 7 categories have positive dominant class imbalance. The insufficient number of negative instances, i.e., positive dominant class imbalance, in these categories is addressed by training the member classifiers using different positive instances and same negative instances. Further lack of sufficient negative instances associated with 3 categories, were the ensemble converged into single classifier, is addressed by over-sampling the negative instances. The highest percentage of over-sampling in this case is 18.23%.

4.4.2 Incremental Sampling Based Learning

The main disadvantage of the ensemble architecture is the expense of sampling, training, testing and maintaining multiple member classifiers. The effectiveness of incremental sampling based learning; another popular and less expensive learning method for classifying very large categories is also examined. The ensemble classifier maintains multiple member classifiers for each iteration. Whereas Incremental sampling based learning, after every iteration, maintains a single classifier only.

Case reduction in incremental sampling is a multi-stage process. This includes training a single classifier on an increasingly larger random subset of cases, observing the trends and stopping when no progress has been made. The subset should take big bites from the original data to ensure the chances of improving performance with more data. The smallest subset should be substantial enough to be the representative of the original dataset and the size of the subset increased gradually to the full sample size. The central theme is to observe the trends and net change in error. A decision on whether further experiment is necessary is made prior to the next increment of dataset size. A significant amount of new data on every iteration should lead to better performance along with an acceptable system complexity. In this procedure, it is important to analyze the cost and achieved benefit before moving to the next iteration of higher sample size. For each classifier, a quality factor defined as a function of normalized value of average F1-Measure is calculated.

The slope in degrees between two consecutive QF's (y-axis) and the normalized average sample size (x-axis) is measured. A slope of zero degrees means there is no improvement in the classifier performance between these two consecutive sample sizes and no further iteration is needed. However, this is an optimal situation that rarely happens. Three sets of experiments have been conducted after fixing predefined threshold of slopes as 1 degrees, 2 degrees and 5 degrees. If the slope in degrees between two consecutive QF's and the normalized average sample size is less than or equal to x degrees, the classifiers formed from the higher sample size is taken as the optimal classifier for the category under consideration. Otherwise the algorithm proceeds to the next iteration.

Here also, occasionally the classifiers may fail to converge for the predefined slope cut-off. The classifier formed by the middle iteration of three consecutive iterations is taken as the optimal classifier, provided the difference of the two consecutive slopes calculated from these three consecutive iterations is less than or equal to 1 degree. Otherwise a single classifier taking complete positive instances and an equal number of negative instances has been trained.

If the total available positive instance of a category is less than 10,000, iteration starts with 10% positive training instances. Similarly, if the total available positive instance of a category is between 10,000 and 100,000 positive instances, iteration starts with 6.25% positive training instances.

For a category, say C_i, of more than 100,000 positive instances, in the 1st iteration of incremental sampling based learning; a single sample is generated taking 2% of the positive instances from C_i and all child categories and an equal number of negative instances from the sibling categories of C_i. Using maximum entropy classification algorithm, a classifier has been trained and average TP Rate and FP Rate after 3-fold cross validation has been calculated. The Quality factor for 1st iteration is calculated.

In the 2nd iteration, another sample is generated taking 4% of the positive instances from C_i and all child categories and an equal number of negative instances from the sibling categories of C_i. The maximum entropy classifier is trained and the Quality factor of the 2nd iteration has been measured. Prior to the third iteration, the achieved benefit after increasing the sample size by 2% in the second iteration is calculated. For this, the slope between two consecutive QF's of 1st and 2nd iteration and average sample size is calculated. If the slope in degrees between two consecutive QF's and average sample size is less than or equal to a predefined threshold, the classifier formed from the higher sample size is taken as the optimal classifier for C_i and incremental sampling based learning terminates for that category. Otherwise the algorithm proceeds to the next iteration of larger sample size.

4.5 Classification of Imbalanced Yahoo! Categories with 100 to 1000 Web Pages

Categories with 100 to 1000 web pages have reasonable numbers of positive instances for classification process. Negative or positive dominant class imbalance is the only machine learning issue associated with the classification of these Yahoo! categories. The class imbalance issue is addressed by either over or under-sampling the negative dataset. This prevents the information loss of positive instances due to sub-sampling. Both these sampling techniques decrease the degree of class imbalance by altering negative dataset distribution. Text content from the body of the web pages is used for classification. Taking complete positive instances and an equal number of negative instances from the sibling categories, 3-fold cross validation is conducted on each category of this group.

4.6 Classification Of the Yahoo! Rare Categories with 10 to 100 Web Pages

The lack of training instances and the negative or positive dominant class imbalance are the main machine learning issues associated with the classification of Yahoo! rare categories of 10 to 100 labeled instances. In the Yahoo! Web directory, the most obvious source of textual information for the purpose of classification is the body of the web pages. However, 22.1% of the web pages have no usable body text. An attempt has been made to reduce the percentage of empty web pages by extending the feature space using the complete textual information within the body, title, meta-keyword and meta-description of the web pages. This decreases the number of web pages with zero textual information. In a web page preprocessing and representation using the body text alone of the rare category web pages, 22.1% of the rare category web pages were excluded from the dataset due to the unavailability of any textual content. However, after extending the web page preprocessing and representation using the textual information from complete web pages, the percentage of empty web pages has been reduced to 18.01%.

Considering the Yahoo! directory rare categories, a slight increase in the number of web pages by extending the feature selection to the complete web page alone will not address the rarity related issues. Hence, data level solutions to address rarity have been examined. Over-sampling is a popular data level approach to address rarity. Over-sampling in its basic form duplicates the rare category dataset. However, crude over-sampling may results into the classifier over fitting. Weiss proposed an adaptive resampling architecture to address the classifier over fitting and related issues associated with random oversampling (Weiss,S.M.,& Indurkhya,N.,1998; Weiss,S.M., Apte, C., Damerau, F.J., Johnson, D.E., Oles, F.J., Goetz, T., Hampp, T.,1999). Effectivenes of this architecture is examined using the Reuters-21578 benchmark data and a real-world customer e-mail routing system. For both dataset, the adaptivly oversampled architecture showed much superior performance. Inspired by this research, we designed a modified version of Weiss adaptive resampling approach to address the rarity associated with Yahoo! Categories of 10 to 100 web pages.

4.6.1 Adaptive Over-Sampling

The basic theory of classifier design using adaptive sampling is to iteratively induce new classifiers by increasing the weight of erroneously classified cases in the training set in the next iteration. Cases having a large error for the current solution are over-sampled with increased frequency. In basic over-sampling, the instances for over-sampling are randomly selected with a probability 1/N, where N is the full sample size. However, in adaptive-over sampling, for each case in the full training set, a record of the current solution performance is kept. In the coming iterations, the instances for over sampling are selected with probability P_i which is determined by the relative probability of the error of the case i.

For each category C_i of 10 to 100 web pages, in the 1st iteration, available positive instances of C_i and all child categories are drawn and combined with an equal number of negative instances from the sibling categories of C_i and a single classifier after 3-fold cross validation has been designed. The average F1-Measure of classification, average recall of C_i and average recall of all child categories are measured.

In the 2nd iteration, the training instances of C_i and all child categories of C_i with lower average recall than a predefined threshold recall are identified and over-sampled by 10%. The test dataset is not over-sampled. The macro-averaged F1-Measure of C_i is calculated. The new average recalls of the C_i and child categories are calculated and used for deciding the over-sampling criteria in the next iteration. If the increase in the F1-Measure of two consecutive iterations is less than 1% no further over-sampling has been performed for the category C_i. Otherwise the algorithm proceeds to the next iteration and repeat the procedure of iteration-2. Two sets of adaptive over-sampling experiments have been conducted by fixing the threshold recall as 85% and 75%. These results are compared with basic over-sampling.

4.7 Conclusion

The methodology of this research has been presented in this chapter. Due to the localized over-abundance of positive instances, rarity and class imbalance within the dataset, any generalized approach for classifying the complete Yahoo! web directory will be highly challenging and inefficient. In this chapter, we designed different architectural solutions to these issues. In the coming chapters, these architectures combined with popular feature selection methods such as Information Gain, Document Frequency and popular classifiers such as Perceptron, Support Vector Machine and Maximum Entropy Classifiers, are examined. The complete Yahoo! web directory of 639,671 categories and 4,140,629 web pages are used to set-up the experiments. Finally, a Yahoo! web directory classification model is designed using the best performing classification technologies.

CHAPTER 5: DIMENSIONALITY REDUCTION OF IMBALANCED DATASETS

5.1 Introduction

In the Yahoo! Web directory, the most obvious source of textual information for the purpose of classification is the body of the web pages. However, 22.1% of the web pages have no usable body text. About 52.4% of web pages contain more than 50 words within the body of the web page, and 25.5% of the web pages contain 1 to 50 words within the body of the web page. Other sources of text are the content in HTML tags including titles, Meta-key word and Meta-Description. However, the amount of text within these tags is relatively very small. Hence we used the textual content from the body of the web pages for the purpose of classification. The following steps have been executed to remove the less informative contents of the textual information within the body segment.

1. Removing HTML tags and scripting languages such as java script
2. Removing stop words
3. Word stemming

The web pages, after preprocessing, contain hundreds of thousands of unique terms. If all the unique terms are used for representing the web pages, the dimension of the feature vectors will be enormous. This results in high time and space complexity for the machine learning algorithm. Hence dimensionality reduction is necessary for web page classification. Dimensionality reduction is also beneficial to reduce the problems of classifier over fitting. Over fitting is the phenomenon where a classifier is tuned to the training data, rather than being generalized from essential characteristics of the training data to classify a new web page.

Dimensionality reduction is popularly achieved by feature selection and feature extraction prior to the classification. While many feature selection techniques have been proposed, a thorough evaluation of these methods over a very large feature space is not reported.

In a broad view, the feature selection criteria can be divided into two sets. One set of feature selection methods, such as, Document Frequency, Mutual Information, Cross Entropy, and Odds Ratio considers the possible value of features that are present in the document. The other set of feature selection methods, such as, Information Gain and Chi-square Statistic, considers all possible values of features including those that are present in and those that are absent from a document. In this chapter, using one representative from each type of feature selection methods, ie, Information Gain and Document frequency feature selection metrics, together with ensemble architecture, two sets of feature selection experiments have been conducted and the relative merits of these feature selection methods are examined. Later, the suitability of these feature selection methods when applied to the content based classification of an imbalanced dataset is analyzed.
The overall goal of this chapter is to address the following query:

> What are the relative merits and demerits of a two-sided feature selection method such as Information Gain and a one-sided feature selection method such as Document Frequency when applied to the content based classification of an imbalanced hierarchical dataset?

The 988 Yahoo! categories of more than 1000 labeled web pages are used to set-up the experiments. The negative instances are drawn from the siblings of each category. A Perceptron classifier is combined with ensemble architecture. The experimental setup used for this research is discussed in Section 5.2. Section 5.3 is the results and discussions. The recommendations of this research, while performing dimensionality reduction of an imbalanced dataset, are summarized in Section 5.4.

5.2 Experiment Setup

While classifying a very large category, say C_i of collective positive instances more than 100,000, in the 1st iteration, 2% of the labeled instances from C_i and all child categories of C_i are drawn and combined with an equal number of negative instances from the sibling categories of C_i. Thus, 50 unique samples of C_i are generated for member classifier design.

In the 2nd, 3rd, 4th, 5th, 6th, 7th, and 8th iterations 4%, 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories are drawn and combined with an equal numbers of negative instances from the sibling categories of C_i. In these iterations, 25, 18, 12, 10, 5, 3 and 2 unique samples of C_i are created for member classifier design.

While classifying categories of 10,000 to 100,000 collective positive instances, in the 1st, 2nd, 3rd, 4th, 5th, and 6th iterations, 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories of C_i are drawn and combined with an equal number of negative instances from the sibling categories of C_i resulting in 18, 12, 10, 5, 3 and 2 unique samples of C_i for member classifier design.

For categories of 1000 to 10,000 positive instances, in the 1st, 2nd, 3rd, and 4th iterations, 10%, 20%, 33.33% and 50% of the positive instances from C_i and all child categories are drawn and combined with equal numbers of negative instances from the sibling categories of C_i resulting in 10, 5, 3, and 2 unique samples of C_i for member classifier design.

The 3-fold cross validation is used for the evaluation. The Information Gain and Document Frequency feature selection methods are applied on the training sets (66.66% instances) of each sample. These feature selection methods process the features independently and assign a numeric score to the features based on some statistical criteria. Using this numeric score, the discriminating features for the classification process, also known as a reduced feature set, are identified. To optimize the dimension of the reduced feature set, two sets of feature selection experiments, taking the highest scoring 20% of the features and 40% of the features of the training set have been conducted and two reduced feature sets for each sample are generated. Using these reduced feature sets, the training sets and test sets of each sample are converted into the reduced document-term matrices.

The main limitation of the discussed feature selection methods is their inability to estimate the effect of co-occurrence of features. For example, two or more features considered independently may not be very effective, but may turn highly effective, when grouped together. This limitation is addressed by applying dimensionality reduction by feature extraction. Feature extraction methods like PCA produce a set of optimum synthetic features of smaller size from the original large feature set without losing any of the significant features.

Feature extraction using PCA is a multi-step procedure. This includes train multiple Perceptron classifiers by varying the number of principal components of the reduced dataset, measure the average performance of each classifier using the projected test dataset, and identify the reduced dataset with the optimum number of principal components producing the best Perceptron classifier performance. By altering the PCA variation factor of the Matlab PCA tool box between 0.005 and 0.05% ten sets of reduced document-term matrices for the training set are generated. Corresponding test sets are also projected into the reduced dimensional space and applied for classification.

Thus, for each sample, for a given feature selection method and a feature selection threshold, ten Perceptron classifiers are trained and tested. The classifier with the highest average F1-Measure, after 3-fold cross validation, has been identified and used as the member classifier for a given sample. Using the average TP Rate and FP Rate of member classifiers together with the variance of the member classifiers, the Quality factor (QF) of the iteration is calculated.

The slope between two consecutive QF's of 1st and 2nd iteration and normalized average sample size are calculated. A slope of zero degree means there is no benefit after increasing the sample size. However, a slope of zero degrees rarely happens. To optimize the sample size and quality factor without affecting the quality of ensemble learning, three sets of experiments for randomly fixed slope thresholds of 1 degrees, 2 degrees and 5 degrees have been conducted. If the slope in degrees between two consecutive QF's and average sample size is less than or equal to a predefined threshold, the member classifiers formed from the higher sample size is considered as the optimal ensemble classifier for C_i and ensemble learning terminates for that category. Otherwise the algorithm proceeds to the next iteration of larger sample size. The optimality of the randomly selected slope cut-offs in the ensemble learning is determined by voting the test sets of each member classifier samples across the entire groups of member classifiers.

5.3 Results and Discussions

The average performance of ensemble Perceptron classifiers for 1, 2 and 5 degrees slope cut-off, combined with information gain and document frequency feature selection methods is summarized in Table 5. For the discussed feature selection methods and feature selection thresholds, a slope cut-off of 1 degree is optimal for Yahoo! web page classification using Perceptron classifier and ensemble architecture. The highest Recall, Precision, and F1-Measure are shown for the Perceptron classifier trained on document frequency features. The variation of the F1-Measure across the hierarchy depth, for various feature selection methods, for 1 degree slope cut-off is as shown in Figure 3.

Theoretically, the knowledge of the feature absence should complement the Perceptron classification process. However, the Perceptron classifier trained on the features identified by Information Gain shows poor classifier performance compared to the Perceptron classifier trained on the features identified by the Document Frequency. This performance drop may be due to the increased sparseness associated with the information gain feature selection method. Variation of sparseness across the hierarchy depth, associated with these feature selection methods is demonstrated in Figure 4. For the discussed feature selection methods, the sparseness increases with the hierarchy depth, resulting in the average performance drop.

Table 5: Macro-Averaged F1-Measure of Ensemble Architecture Combined with Perceptron Classifiers

Perceptron Classifier trained with	**1 degree cut-off**	**2 degree cut-off**	**5 degree cut-off**	**Average sparseness for 1 degree cut-off (%)**
highest scoring 20% Information Gain features after PCA based feature extraction	66.07	54.43	38.46	42.23
highest scoring 40% Information Gain features after PCA based feature extraction	49.16	40.18	30.67	50.82
highest scoring 20% Document Frequency features after PCA based feature extraction	85.26	71.96	59.02	21.19
highest scoring 40% Document Frequency features after PCA based feature extraction	66.94	55.55	44.18	37.93

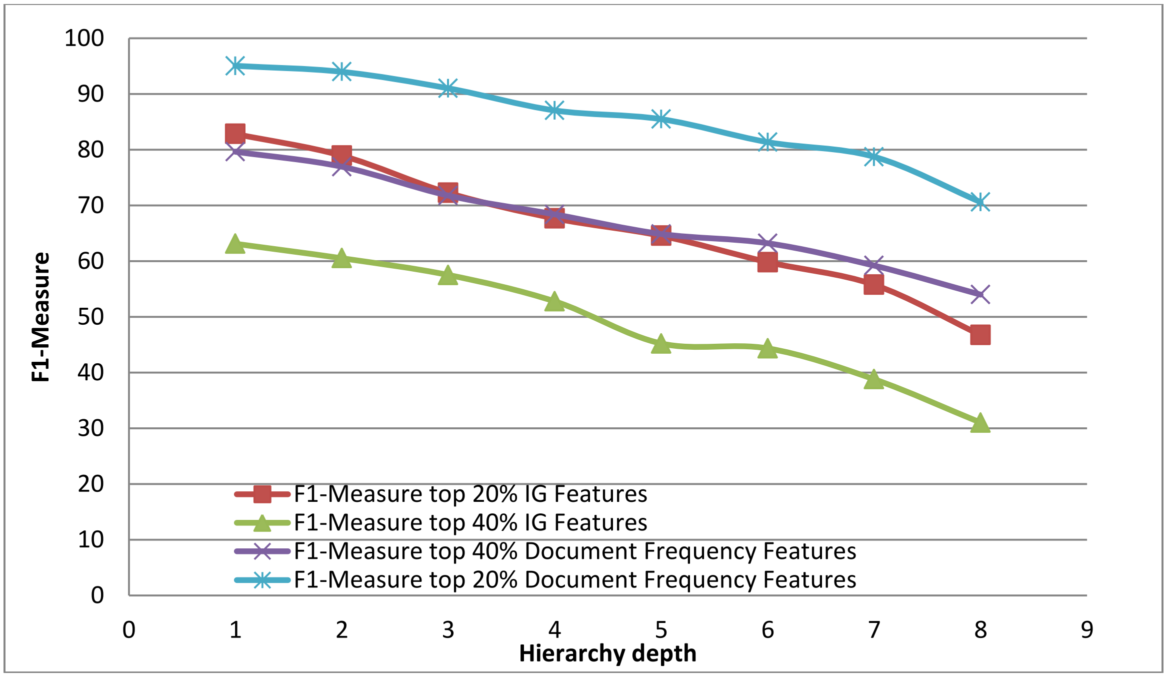

Figure 3: Variation of F1-Measure with Hierarchy Depth for Ensemble Architecture Combined with Perceptron Classifiers

For the top 20% features, when switched from the document frequency to the information gain feature selection method, there is a 21.14% increase in the sparseness of the document-term matrix. Considering the advantages of the Perceptron classifiers trained on Document Frequency features over the Perceptron classifiers trained on Information Gain feature selection and the suitability of the document frequency feature selection method for large-scale web page classification problem, the Information Gain feature selection method is not used in the rest of this research.

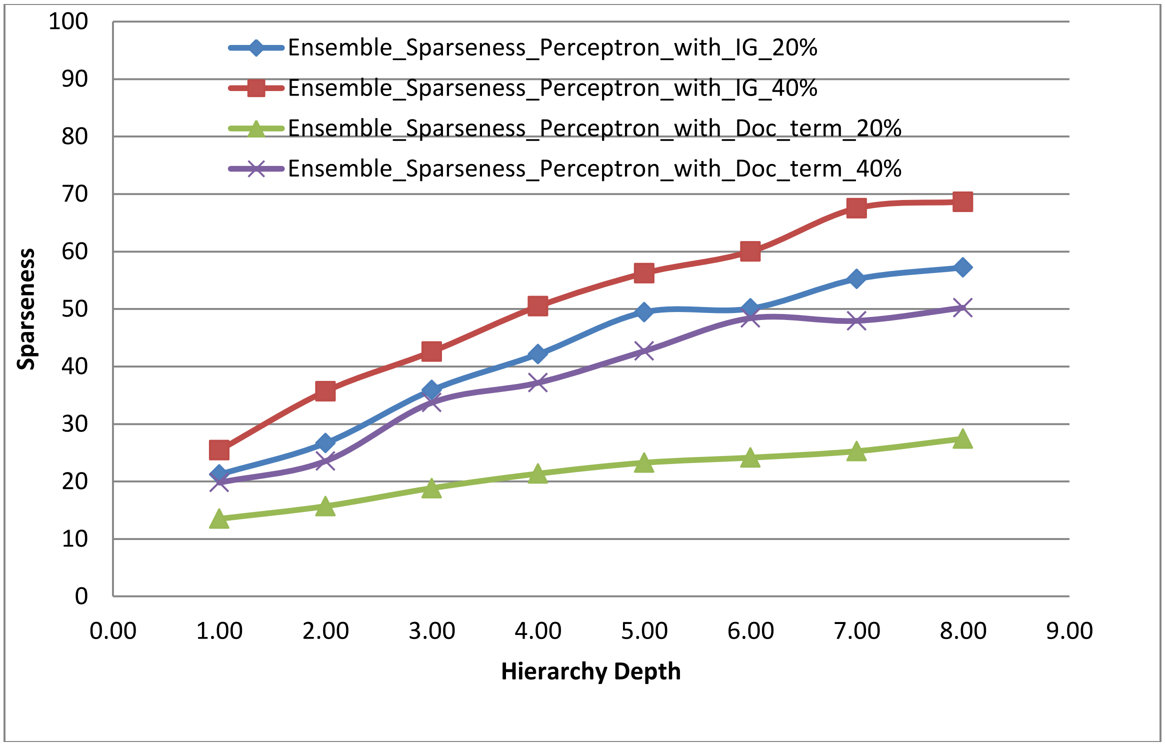

Figure 4: Sparseness with Hierarchy Depth for Ensemble Architecture Combined with Perceptron Classifiers

The Perceptron classifier is designed using Matlab 'newff' routine. While designing a Perceptron network object using the newff function, the hidden layer array must be a single output layer with a single neuron. The tan-sigmoid transfer function and conjugate gradient training function have been used for the Perceptron design. To train the network, the 'train' routine is used. Normally, training of a Perceptron classifier stops when any of these conditions occurs: the maximum number of epochs (repetitions) is reached, the performance gradient falls below min_grad, or the performance is minimized to the goal. However, while training the Perceptron, the first two parameters have been adjusted appropriately until the best performance is reached.

5.4 Conclusions of the Feature Selection Experiments

Web directories generally have a skewed category distribution and unbalanced class distribution. The stratified sampling of the negative instances from multiple sibling categories is performed to address over-abundance of the negative instances and associated class imbalance. This sampling scheme makes the prior probability distribution of the negative features extremely small compared to that of the positive features. Thus, the higher value gained by a feature by applying feature selection methods such as Information Gain might be due to feature absence rather than feature presence. This imposes more sparseness in the reduced feature space. Sparseness of the dataset unfavorably affects statistical methods such as PCA based feature extraction. For an improved classifier performance, the degree of sparseness in the reduced feature space should be minimum. Compared to two-sided feature selection method such as Information Gain, this can be easily achieved by one-sided feature selection method such as Document Frequency.

Moreover, there is a visible drop in the average classifier performance with the hierarchy depth. Most of the earlier research on hierarchical dataset also reported this observation. They concluded that the performance drop in lower hierarchies is due to the lack of sufficient training instances. However, all categories used in the discussed set of experiments have more than 1000 positive instances. The observed performance drop associated with the lower hierarchies points to the limitations of the discussed feature selection methods or Perceptron classifier. To make this clear, another set of experiments are conducted on the same samples using one-class SVM and two-class SVM classifiers.

Document Frequency is applied for feature selection. Similar to the Perceptron classifiers, the SVMs also show a drop in the average performance with hierarchy depth. The results of those experiments are discussed in detail in Chapter 6.

The poor classifier performance with hierarchy depth is related to the increased average sparseness of the reduced feature space (Figure 4). As categories become more specific with hierarchy depth, the ability of the feature selection methods to identify the discriminating features applicable to the majority of the web pages may be decreasing, resulting in an increased average sparseness and poor classifier performance. Feature selection using subspace clustering or genetic programming may be more appropriate to address this situation.

Since sparseness negatively affects the classifier performance, a classification that requires feature selection and feature extraction prior to the classification can be highly challenging to classify the rare categories of web directories. This is because the basic over-sampling methods used to address rarity is not introducing new data.

CHAPTER 6: CLASSIFICATION OF VERY LARGE AND HIGHLY IMBALANCED WEB DIRECTORIES

6.1 Introduction

In recent years, a large number of statistical learning methods have been applied to the web page classification problem. Since a large number of methods and results are available, a cross-method evaluation is important to comprehend the current status of the web page categorization research. The comparison of different text and web page classification methods, however, is very difficult due to the absence of a cohesive methodology for the matter-of-fact evaluation. Cross-method comparisons with a limited number of methodologies have been reported in the literature. However, these types of small-scale comparisons can either lead to highly comprehensive statements that are based on inadequate observations, or provide limited insight into the challenges of real time web page classification.

The lack of a standard data collection is the main bottle-neck for cross-method comparison in web page categorization research. For a given dataset, there are many possible ways to introduce inconsistent variations. Whether the reported classifier performance on different versions of a dataset is comparable is not clear. Incomparability across different evaluation measures used in individual experiments is another concern on cross-experiment evaluation. In general, one should be highly vigilant while comparing the published text categorization research. Due to the aforementioned issues, a comprehensive evaluation of different web page classification methods using consistent samples is not reported.

This research, so far, has determined that the document frequency feature selection method with PCA based feature extraction and ensemble architecture has given the best result for the classification of categories with more than 1000 web pages. This architecture is based on the Perceptron classifier. In this chapter, using the same sample and architecture, comparisons of popular statistical learners and maximum entropy models are conducted and their relative merits and demerits are discussed. The distribution of very large categories across the hierarchy depth is as shown in Table 6.

Table 6: The Distribution of Very Large Yahoo! Categories across the Hierarchy Depth

level	Total categories	Categories with 1000 to 10,000 web pages	Categories with 10,000 to 100,000 web pages	Categories with 100,000 to 600,000 web pages
1	14	0	0	14
2	133	0	92	41
3	233	45	188	0
4	347	227	120	0
5	138	106	32	0
6	87	84	3	0
7	33	33	0	0
8	3	3	0	0

The overall goals of this chapter are to address the following queries:

1. What is the average performance of popular statistical and maximum entropy based classifiers when applied to the content based classification of a dataset?
2. Whether one-class learning is a better alternative for class imbalance?

The experimental setup used for this research is discussed in Section 6.2. Section 6.3 is the results and discussions. Summary of this chapter is Section 6.4.

6.2 Experiment Setup

This research, so far, has determined that the document frequency feature selection method with PCA based feature extraction and ensemble architecture has given the best result for the classification of categories with more than 1000 web pages. This architecture is based on the Perceptron classifier. Now we compare the effectiveness of the two-class SVM classifier by replacing the Perceptron classifier with the SVM classifier in the same architecture.

Later, taking the positive instances of the same samples and repeating document frequency based feature selection and PCA based feature extraction experiments, the effectiveness of one-class learning is examined using the popular one-class SVM classifier. In one-class learning, the member classifiers of the ensembles are trained using the positive instances only and tested using an equal number of positive and negative instances. Here the origin is treated as the member of the second class and the candidate class is separated from the origin. Hence, we expect that the misconceptions on learning the negative dataset will be alleviated.

The extensive feature selection and feature reduction, prior to the classification, is the main limitation of Perceptron and SVMs when applied to large-scale web page classification. Moreover, to this point, this research concludes that the average performance of the popular feature selection methods decreases with hierarchy depth resulting in the poor classifier performance.

The maximum entropy classifier, when used for web page classification, does not require feature selection or feature extraction prior to the classification. While designing a maximum entropy model one should prefer the most uniform model satisfying any given constraint. For example, consider a four way web classification task where on an average 40% of the documents with the word “internet” is in the “computer and internet” class. Given a document with “internet” in it, we would say it has a 40% chance of being a “computer and internet” document and a 20% chances for each of the other three classes. If a document does not have “internet” we will guess a uniform class distribution, 25% each. The MEGA Model Optimization Package is used to implement the maximum Entropy Classifier (*MEGA Model Optimization Package,2007*).

The average performance of ensemble SVM classifiers for a 1, 2 and 5 degrees slope cut-off, combined with the document frequency feature selection method and PCA based feature extraction is summarized as Table 7. LIBSVM package is used to set up these experiments. The variation of the F1-Measure across the hierarchy depth, for one-class SVM and two-class SVM as is shown in Figure 5.

Table 7: Macro-Averaged F1-Measure of Ensemble Architecture Combined with SVM

Classifier	1 degree cut-off	2 degree cut-off	5 degree cut-off
Two-class SVM trained on the highest scoring 20% features	81.45	73.25	58.04
Two-class SVM trained on the highest scoring 40% features	61.53	53.64	42.72
One-class SVM trained on the highest scoring 20% features	70.61	61.34	48.13
One-class SVM trained on the highest scoring 40% features	55.75	46.68	37.51

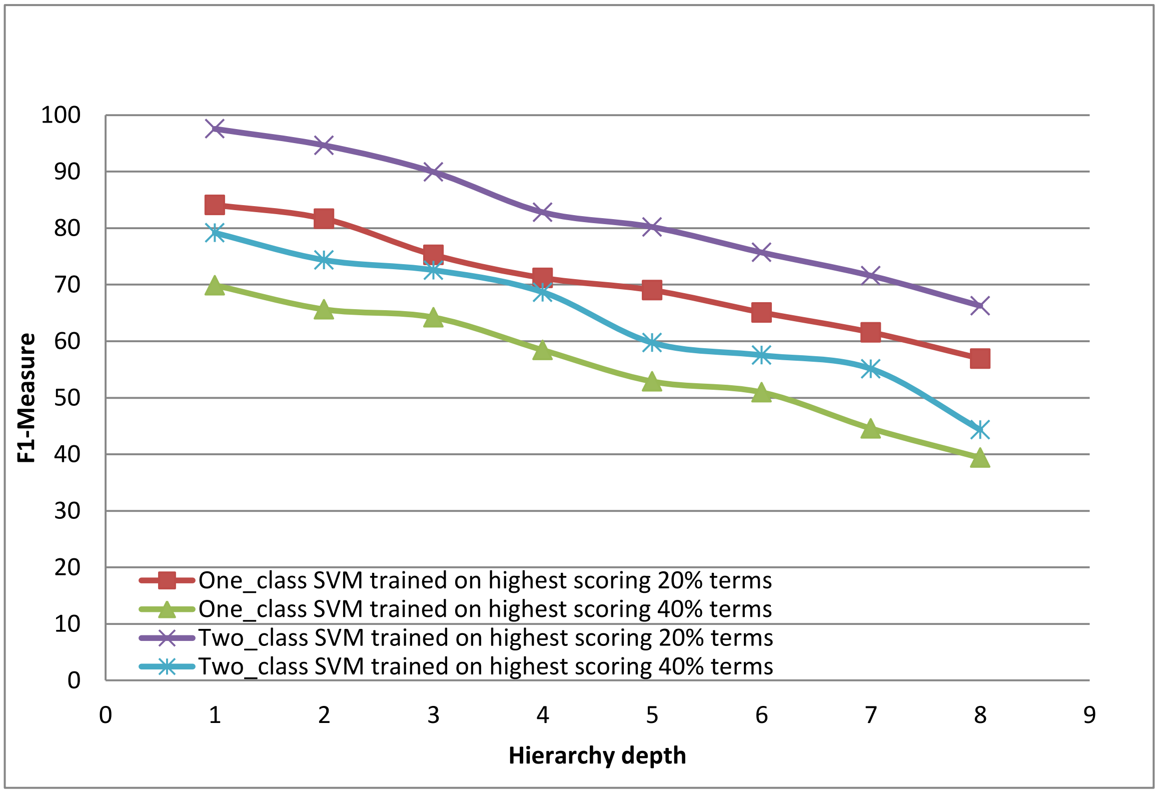

Figure 5: Variation of F1-Measure with Hierarchy Depth for Ensemble Architecture Combined with SVMs

The variation of average F1-Measure with the hierarchy depth for the ensemble architecture combined with Maximum Entropy Classifier is as shown in Figure 6. The Macro-averaged F1-Measure for 1 degree slope cut-off is 87.86%. The macro-averaged rare category recall for 1, 2 and 5 degree cut-offs for an ensemble architecture combined with maximum entropy classifier is 80.44% and 66.63% and 48.89% respectively. The average performance of ensembles of Maximum Entropy classifiers for 1, 2 and 5 degrees slope cut-off is summarized as Table 8.

Table 8: Average Performance of Ensemble Architecture Combined with Maximum Entropy Classifier

Slope Cut-off	F-Measure	Rare Category Recall
1 degree	87.86	80.44
2 degree	74.39	66.63
5 degree	55.43	48.89

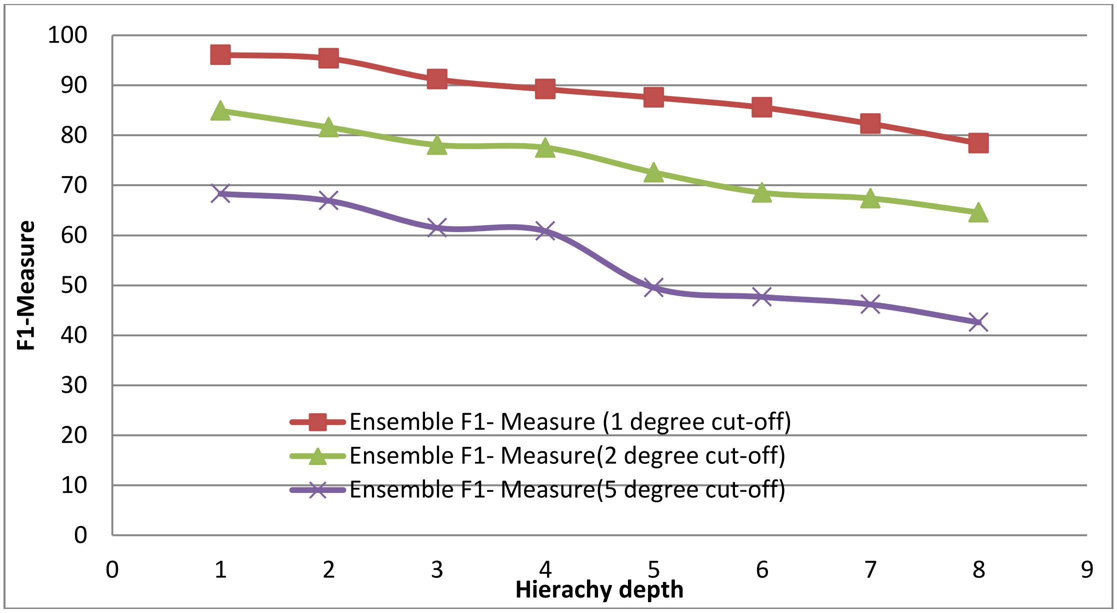

Figure 6: Variation of F1-Measure with Hierarchy Depth For Ensemble Architecture Combined with Maximum Entropy Classifiers

6.3 Results and Discussion

The relative merits and demerits of Perceptron classifier, SVM classifiers and Maximum entropy classifier combined with ensemble architecture, when applied to very large and highly imbalanced dataset classification are discussed below. To avoid the ambiguity in classifier evaluation due to the inconsistent variations in the dataset, we conducted all experiments using the same samples, training set and test set. Over-all performance of these classification algorithms in association with ensemble architecture is summarized in Table 9. The highest average performance is shown for the Maximum Entropy classifier.

Table 9: Average Performance of Ensemble Architecture Combined with Popular Classification Techniques When Applied to Very Large Datasets

Classifier	**Average F1-Measure**	**Average Recall**	**Average Precision**	**Average recall of child categories with 1 to 10 web pages**
Maximum Entropy Classifier	87.86	86.21	89.56	80.44
Perceptron with top 20% features of highest Document Frequency.	85.26	85.9	84.63	79.45
Perceptron with top 20% feature of highest Information Gain	66.07	67.06	65.08	57.59
One-Class SVM with top 20% features of highest Document frequency	70.61	67.83	73.63	55.12
Two-class SVM with top 20% features of highest Document Frequency	81.45	81.92	80.99	75.41

The last column of Table 9 (average recall of child categories with 1 to 10 web pages) indicates the average performance of the large-scale classification methodology while classifying the rare child category web pages representatives within the dataset that has been recursively assigned into it. The rare category performance is separately analyzed to ensure that the applied large-scale web page classification architecture maintains a reasonable quality of performance while classifying the rare child category representative within the dataset. A separate rare category evaluation is important due to the following reasons. Firstly we recursively assigned the web pages of the child categories into the parent categories to decrease the degree of rarity. Moreover, in a classification problem, a rare child category of a very large parent category may be more interesting making the misclassification of the rare category web pages more expensive. Most of the earlier large-scale hierarchical web page classification research work; the rare category performance is not clear making a conclusion on the average performance of the applied technologies towards rare child category classification very difficult.

To check the significant difference in the average F1-Measures between different classification methodologies if any, a One-Variable Chi-Square test of independence is conducted using macro-averaged F1-Measure. The formula for Chi-square test of independence is, $X^2=\sum (O-E)^2 / E$, where O is the observed frequency, and E is the expected frequency

The degree of freedom for the one-dimensional chi-square statistic is defined as,
df = (C - 1) where C is the number of levels of the variable. In our case, the degree of freedom is 5-1= 4. The null hypothesis, H_0 and alternative hypothesis, H_1 are as follows:

H_0: There is significant difference in the average performance of the different classification algorithms.

H_1: There is no significant difference in the average performance of the different classification algorithms**.**

The alpha level is set as 0.025. The critical value for Chi-square for 0.025 level and degree of freedom 4 is 11.143. Hence, the null hypothesis will be accepted if the obtained chi-square value is less than 11.143. In this case, the obtained sum of chi-square value is 4.58 (Table 10). Hence we conclude that there is significant difference in the average performance of the different classification algorithms.

Table 10: One-Variable Chi-Square Test on the Macro-Averaged F1-Measure of Different Classification Methodologies

Attribute	F1-Measure	chi-square value
F1-Measure of Maximum Entropy Classifier	87.86	1.180219
F1-Measure of Perceptron with top 20% features of highest Document Frequency.	85.26	0.627988
F1-Measure of Perceptron with top 20% feature of highest Information Gain	66.07	1.895877
F1-Measure of One-Class SVM with top 20% features of highest Document frequency	70.61	0.745937
F1-Measure of Two-class SVM with top 20% features of highest Document Frequency	81.45	0.130863
Total		4.58

The Mean Absolute Deviation (MAD) of the macro-averaged F1-Measure with hierarchy depth, for the discussed classification techniques is shown in Table 11. The higher the MAD, the greater will be the dispersion of the F1-Measure with hierarchy depth. Compared to the SVM's and Perceptron trained after feature reduction, the dispersion is low for the Maximum Entropy Classifiers.

The highest MAD and lowest average performance is associated with the Perceptron combined with Information Gain Feature selection method. Thus compared to statistical learners that require feature selection and feature extraction prior to the classification, the

Maximum Entropy classifier shows more consistent performance across the hierarchy depth.

Table 11: MAD of F1-Measure with hierarchy depth for ensemble architecture

Classification techniques	MAD	Macro-F1-Measure
Ensemble learning (Max. Entropy Classifier)	4.67	88.33
Ensemble learning (Perceptron+Doc. Freq)	6.39	85.42
Ensemble learning (One-class SVM+Doc. Freq.)	8.9	70.61
Ensemble learning (Two-class SVM+Doc. Freq)	7.45	82.34
Ensemble learning (Perceptron + IG)	9.35	66.08

Considering the advantages of the Maximum Entropy classification algorithm over the Perceptron and SVM classification methods and the suitability of the Maximum Entropy classification algorithm for large-scale web page classification problem, the Maximum Entropy classification algorithm combined with an appropriate architecture to address class imbalance, rarity and over-abundance of positive instances is used for remaining experiments.

6.4 Conclusions

A comparison of the average performance of the Perceptron trained on Document Frequency and Information Gain, one-class SVM, two-class SVM and Maximum Entropy Classification model has been conducted. The Yahoo! categories of more than 1000 labeled instances, is used to set up the experiments. Later, the relative merits and demerits of Perceptron classifier, SVM classifiers and Maximum entropy classifier, when applied to very large and highly imbalanced dataset classification are discussed.

One-class learning is considered as a popular algorithmic solution to class imbalance. However, the effectiveness of one-class learning when applied to very large datasets similar to web directories is not reported in the literature. In this research, compared to two-class SVM and Perceptron, the one-class SVM showed the lowest classification performance.

Similarly, compared to the Perceptron and SVM's, the Maximum Entropy classifier shows the highest F1-Measure of 87.86%. The Maximum Entropy classifier, when used for web page classification, does not require feature selection or feature extraction prior to the classification. Hence maximum entropy models are free from the bias of the popular feature selection methods discussed in Chapter 5. This is the reason for the improved average performance of the Maximum Entropy Classifiers. Moreover, compared to statistical learners that require feature selection and feature extraction prior to the classification, the Maximum Entropy classifier shows more consistent performance across the hierarchy depth. These properties make the Maximum Entropy classifier more suitable for large scale web page classification than the Perceptron, one-class SVM and two-class SVM classifiers.

CHAPTER 7: THE MACHINE LEARNING ARCHITECTURE FOR IMBALANCED DATASET CLASSIFICATION

7.1 Introduction

The class imbalance, rarity and large-sample learning issues within the web directories make applying feature reduction techniques and classification algorithms very difficult. For example, in the Yahoo! web directory, 0.16% of the total categories (988 categories) are very large, containing 1000 to 600,000 labeled web pages of their own; whereas, 19.06% of the Yahoo! categories are absolutely rare categories of 10 to 100 labeled web pages. Classification algorithms, when applied to very large categories of more than 1000 labeled instances should address the machine learning issues due to the class imbalance and large-sample learning. Conversely, classification algorithms, when applied to rare categories of 10 to 100 labeled web pages should address the machine learning issues due to class imbalance and rarity. Another 1.58% of Yahoo! categories are reasonably sized categories holding 100 to 1000 labeled web pages. However, the abundance or shortage of negative instances in the sibling categories makes these categories imbalanced.

The impact of a large training space due to the over-abundance of positive instances, class imbalance and absolute rarity during the different stages of the classification process should be prevented or addressed. This chapter investigates the effectiveness various architectural solutions to these issues.

The overall goals of this chapter are to address the following queries:

1. Which architecture is more appropriate for learning very large categories of 1000 to 600,000 labeled instances: Incremental sampling based learning or Ensemble learning?
2. Which architecture is more appropriate for learning rare categories: adaptive over sampling or crude over sampling?
3. How class imbalance within the dataset can be effectively addressed?

7.2 Comparison of Ensemble Learning and Incremental Sampling Based Learning for Classifying Very Large and Imbalanced Datasets

S far, in this research, all experiments are conducted on Yahoo! categories of more than 1000 web pages. An ensemble architecture discussed in Chapter 4 is used to set up the experiments. The main disadvantage of the proposed ensemble architecture is the expense of sampling, training, testing and maintaining multiple member classifiers. The effectiveness of Incremental Sampling based learning; another popular and less expensive active learning method for classifying very large categories is also examined. The experimental set-up for incremental sampling based learning is discussed in Chapter 4. Considering the advantages of Maximum Entropy Classifiers over the Perceptron and SVMs, the Maximum Entropy Classifiers are used to set-up the incremental sampling based learning experiments.

Similar to ensemble learning, case reduction in incremental sampling is a multi-stage process. However, a single classifier is maintained in each iteration. This includes training a single classifier on an increasingly larger random subset of cases, observing the trends and stopping when no progress has been made. The subset should take big bites from the original data to ensure the chances of improving performance with more data. The smallest subset should be substantial enough to be the representative of the original dataset and the size of the subset increased gradually to the full sample size. The central theme is to observe the trends and net change in error. A decision on whether further experiment is necessary is made prior to the next increment of dataset size. A significant amount of new data on every iteration should lead to better performance along with an acceptable system complexity. In this procedure, it is important to analyze the cost and achieved benefit before moving to the next iteration of higher sample size. For each classifier, a quality factor defined as a function of normalized value of average F1-Measure is calculated.

The slope in degrees between two consecutive QF's (y-axis) and the normalized average sample size (x-axis) is measured. A slope of zero degrees means there is no improvement in the classifier performance between these two consecutive sample sizes and no further iteration is needed. However, this is an optimal situation that rarely happens. Two sets of experiments are conducted after fixing predefined threshold of slopes as 1 degree and 2 degrees. If the slope in degrees between two consecutive QF's and the normalized average sample size is less than or equal to x degrees, the classifiers formed from the higher sample size is taken as the optimal classifier for the category under consideration. Otherwise the algorithm proceeds to the next iteration.

In a large scale classification application with hundreds of categories, occasionally the classifiers may fail to converge for the predefined slope cut-off. The classifier formed by the middle iteration of three consecutive iterations is taken as the optimal classifier, provided the difference of the two consecutive slopes calculated from these three consecutive iterations is less than or equal to 1 degree. Otherwise a single classifier taking complete positive instances and an equal number of negative instances has been trained.

For a category, say C_i, of more than 100,000 positive instances, in the 1st iteration of incremental sampling based learning; a single sample is generated taking 2% of the positive instances from C_i and all child categories and an equal number of negative instances from the sibling categories of Ci. Using maximum entropy classification algorithm, a classifier has been trained and average TP Rate and FP Rate after 3-fold cross validation has been calculated. The Quality factor for 1st iteration is calculated.

In the 2nd iteration, another sample is generated taking 4% of the positive instances from C_i and all child categories and an equal number of negative instances from the sibling categories of C_i. The maximum entropy classifier is trained and the quality factor of the 2nd iteration has been measured. Prior to the third iteration, the achieved benefit after increasing the sample size by 2% in the second iteration is calculated. For this, the slope between two consecutive QF's of 1st and 2nd iteration and average sample size is calculated. If the slope in degrees between two consecutive QF's and average sample size is less than or equal to a predefined threshold, the classifier formed from the higher sample size is taken as the optimal classifier for C_i and incremental sampling based learning terminates for that category. Otherwise the algorithm proceeds to the next iteration of larger sample size.

In the 3rd, 4th, 5th, 6th, 7th, and 8th iterations, to generate the sample, 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories are drawn and combined with an equal numbers of negative instances from the sibling categories of C_i.

While classifying categories of 10,000 to 100,000 collective positive instances, in the 1st, 2nd, 3rd, 4th, 5th, and 6th iterations, 6.25%, 8.33%, 10%, 20%, 33.33% and 50% of the labeled instances of C_i and all child categories of C_i are drawn and combined with an equal number of negative instances from the sibling categories of C_i.

For categories of 1000 to 10,000 positive instances, in the 1st, 2nd, 3rd, and 4th iterations, 10%, 20%, 33.33% and 50% of the positive instances from C_i and all child categories are drawn and combined with equal numbers of negative instances from the sibling categories of C_i.

The variation of macro-averaged F1-Measure across the hierarchy depth for 1 degree and 2 degree slope cut-offs is as shown in Figure 7. The average performance of Incremental sampling based learning combined with Maximum Entropy classifiers for 1 and 2 degrees slope cut-off is summarized as Table 12.

Table 12: Average Performance of Incremental Sampling Based Learning

Slope Cut-off	F-Measure	Rare Category Recall
1 degree	79.54	71.07
2 degree	71.16	62.77

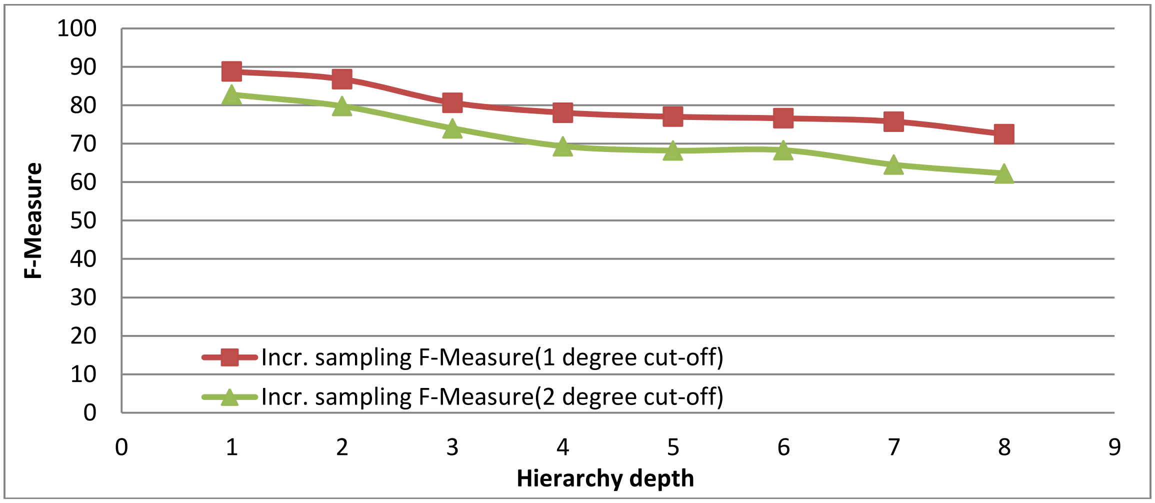

Figure 7: Variation of F1-Measure with Hierarchy Depth for Incremental Sampling Based Learning

A comparison of the average performance of the Maximum Entropy classifier combined with ensemble architecture and incremental sampling based learning for 1 degree slope cut-off is summarized as Table 13.

Compared to ensemble architecture, there is a significant performance drop in the classifier performance associated with the incremental sampling based learning architecture. This is because of the information loss due to the sub-sampling associated with the incremental sampling based learning. Hence this research consider the ensemble architecture as the optimal architectureal solution for very-large dataset classification.

Table 13: Comparison of Ensemble Architecture and Incremental Sampling Based Learning When Applied to Very Large Datasets

Architecture	**Recall**	**Precision**	**F1-Measure**	**Recall of the rare categories with 1 to 10 web pages**
Ensemble	86.21	89.57	87.86	80.44
Incremental Sampling	77.77	81.39	79.54	71.07

7.3 An Evaluation of Focused Under-Sampling and Over-Sampling to Address Class Imbalance Associated with Categories of 100 to 1000 Labeled Instances.

The explicit machine learning issue associated with the Yahoo! categories of 100 to 1000 web pages is negative and positive dominant class imbalance. However, the chances of rarity and over-abundance of positive instances associated with border categories of this group cannot be ignored.

This research has already been examined the effectiveness of different class imbalance handling techniques. This includes various sub-sampling techniques and one-class learning. These class imbalance techniques are tailored in the ensemble and incremental sampling based learning architectures, modeled to classify Yahoo! categories of more than 1000 positive instances.

In incremental sampling based learning, sub-sampling of the positive instances is performed to address the over-abundance of positive instances. The class imbalance is addressed by sub-sampling the negative instances.

In ensemble learning, the class imbalance is addressed by sub-sampling the negative instances. The over-abundance of positive instances is addressed by maintaining multiple member classifiers. In our proposed ensemble model, the complete positive instances are distributed across multiple member classifiers and hence there is no information loss due to the sub-sampling of the positive instances.

A comparison of these architectures unseals the relative merits and demerits of sub-sampling and one-class learning as class imbalance handling practices. In this research, compared to ensemble learning, incremental sampling based learning showed an inferior average performance. This is due to the sub-sampling of the positive instances associated with the incremental sampling based learning. Compared to one-class learning, sub-sampling of negative instances is much superior to address class imbalance within the dataset.

Considering these observations, we propose sub-sampling of the negative instances to address the class imbalance associated with categories of 100 to 1000 web pages. As the next step, the proposed class imbalance handling technique is tailored on an appropriate machine learning architecture and used further to classify the Yahoo! categories of 100 to 1000 labeled instances. Before fixing the machine learning architecture, the chances of rarity and overabundance of positive instances associated with the Yahoo! categories of around 100 and 1000 web pages are examined.

If the ensemble learning turns unprofitable for the given Yahoo! subcategory of more than 1000 labeled instances, our ensemble model converges into a single classifier and undergoes three fold cross validation. Of the total 988 Yahoo! categories of more than 1000 positive instances, 24 categories converged like this into the single classifiers. This covers 89.56% of the categories of less than 1700 positive instances.

Similarly, in the adaptive over-sampling based learning architecture designed to classify rare categories with 10 to 100 web pages, all categories of more than 75 web pages is oversampled by less than or equal to 20% only. Whereas, all categories of less than 45 web pages is oversampled by more than 50%. The highest over-sampling is performed with the Yahoo! categories of less than 20 labeled web pages.

Based on these observations, there are no evidences for rarity or over-abundance of positive instances associated with Yahoo! categories of 100 to 1000 web pages. Hence, for each category of this group, a three-fold cross validation taking the complete positive instances and equal number of negative instance using the best performing Maximum entropy classification algorithm will meet the machine learning architectural requirements.

Text content from the body of the web pages is used for classification. The Maximum Entropy Classifiers are used to set up the experiments. A 3-fold cross validation is conducted on each category of this group. Average recall and precision of this group of categories is 73.02% and 79.65% respectively. Variation of rare category performance and F1-Measure across the hierarchy depth is demonstrated in Figure 8 Average F1-Measure is 76.19%. Average recall of the rare categories is 67.49%.

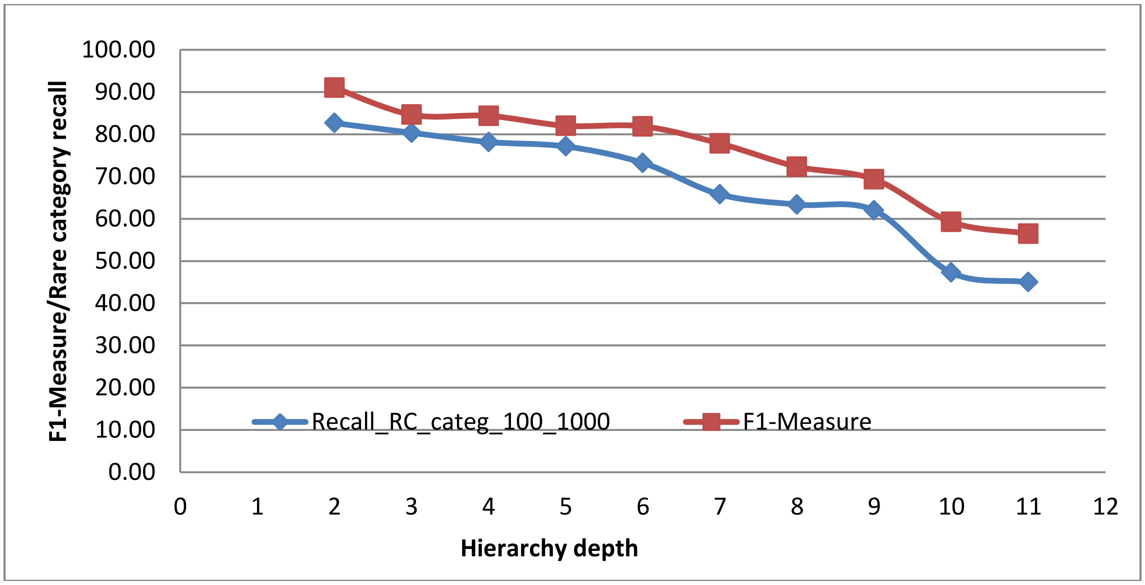

Figure 8: Average Classifier Performance of Yahoo! Categories with 100 to 1000 Labeled Instances

7.4 An Evaluation of Adaptive Over-Sampling to Address Rarity

The lack of training instances and the negative or positive dominant class imbalance are the main machine learning issues associated with the classification of Yahoo! rare categories of 10 to 100 labeled instances. In the Yahoo! Web directory, the most obvious source of textual information for the purpose of classification is the body of the web pages. However, 22.1% of the web pages have no usable body text. An attempt has been made to reduce the percentage of empty web pages by extending the feature space using the complete textual information within the body, title, meta-keyword and meta-description of the web pages. This decreases the number of web pages with zero textual information. In a web page preprocessing and representation using the body text alone of the rare category web pages, 22.1% of the rare category web pages were excluded from the dataset due to the unavailability of any textual content. However, after extending the web page preprocessing and representation using the textual information from complete web pages, the percentage of empty web pages has been reduced to 18.01%.

Considering the Yahoo! directory rare categories, a slight increase in the number of web pages by extending the feature selection to the complete web page alone will not address the rarity related issues. Hence, data level solutions to address rarity have been examined. Over-sampling is a popular data level approach to address rarity. Over-sampling in its basic form duplicates the rare category dataset. However, crude over-sampling can be insufficient. In this research, advanced over-sampling of the rare category web pages, adaptive over-sampling, is experimented to address the rarity.

7.4.1 Adaptive Over-Sampling

The basic theory of classifier design using adaptive sampling is to iteratively induce new classifiers by increasing the weight of erroneously classified cases in the training set in the next iteration. Cases having a large error for the current solution are over-sampled with increased frequency. In basic over-sampling, the instances for over-sampling are randomly selected with a probability 1/N, where N is the full sample size. However, in adaptive-over sampling, for each case in the full training set, a record of the current solution performance is kept. In the coming iterations, the instances for over sampling are selected with probability P_i which is determined by the relative probability of the error of the case i.

For each category C_i of 10 to 100 web pages, in the 1st iteration, available positive instances of C_i and all child categories are drawn and combined with an equal number of negative instances from the sibling categories of C_i and a single classifier after 3-fold cross validation has been designed. The average F1-Measure of classification, average recall of C_i and average recall of all child categories are measured.

In the 2nd iteration, the training instances of C_i and/or all child categories of C_i with lower average recall than a predefined threshold recall are identified and over-sampled by 10%. However, the test dataset is not over-sampled. The macro-averaged F1-Measure of C_i is calculated. The new average recalls of the C_i and child categories are calculated and used for deciding the over-sampling criteria in the next iteration. If the increase in the F1-Measure of two consecutive iterations is less than 1% no further over-sampling has been performed for the category C_i. Otherwise the algorithm proceeds to the next iteration and repeat the procedure of iteration-2. Two sets of adaptive over-sampling experiments have been conducted by fixing the threshold recall as 85% and 75%. These results were compared with basic over-sampling.

The highest average F1-Measure is shown for recall cut-off of 85%. Later, a basic over-sampling experiment is conducted for the same cut-off recall. Here, instead of measuring the average recall of child categories of C_i separately and over-sampling them separately, average recall of C_i together with child documents is measured. If this recall is less than 85%, 10% of web pages of training set is randomly identified and duplicated. The average improvement in the total F1-Measure on every iteration for different recall cut-offs after every iteration is demonstrated in Figure 9. The percentage of over-sampling and the percentage of the new categories over-sampled after every iteration for recall cut-offs of 75% and 85% are shown in Table 14 and Table 15.

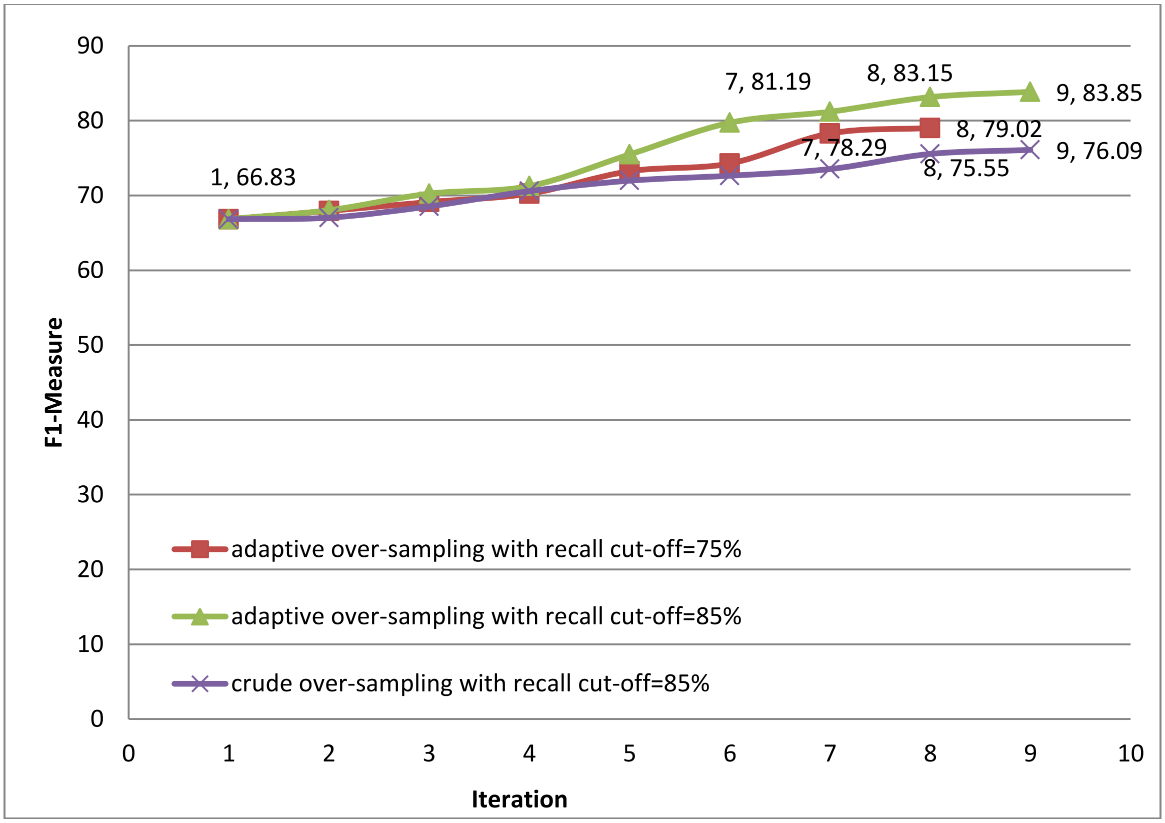

Figure 9: Comparison of Percentage of Over-Sampling and Average Classifier Performance for Different Adaptive Over-Sampling and Crude Over-Sampling Experiments

Table 14: Adaptive Over-Sampling Statistics for 75% Recall Cut-Off

iteration #	Highest % of over-sampling	% sub-categories involved in the over-sampling	% of new sub-categories involved in over-sampling	F1-Measure
0	0	0	0	66.83
1	10	41.44	41.44	67.94
2	20	39.69	15.73	69.12
3	30	25.89	8.91	70.25
4	40	17.98	1.73	73.24
5	50	12.62	1.02	74.27
6	60	8.77	0.83	78.29
7	70	3.09	0.05	79.02
		Average over-sampling =18.68%		

Table 15: Adaptive Over-Sampling Percentage for 85% Recall Cut-Off

iteration #	Highest % of over-sampling	% sub-categories involved in the over-sampling	% of new sub-categories	F1-Measure
0	0	0	0	66.83
1	10	60.72	60.72	68.03
2	20	54.29	37.23	70.24
3	30	48.9	22.19	71.25
4	40	40.17	18.37	75.48
5	50	26.15	12.01	79.72
6	60	17.67	10.28	81.19
7	70	14.19	5.42	83.15
8	80	10.57	3.23	83.85
		Average over-sampling=32.76%		

The highest average F1-Measure is shown for a recall cut-off of 85%. The average percentage of over-sampling associated with 85% recall cut-off is 32.76% whereas the average percentage of over-sampling associated with 75% recall cut-off is 18.68%. The average F1-Measure for 75% and 85% recall cut-off is 83.85% and 79.02%. An achieved benefit of 4.83% in the average F1-Measure, associated with 14.08% increase in the average over-sampling for a recall cut-off of 85%, is questionable. Especially, an exact copy of samples after over-sampling may lead to the classifier over-fitting. To prevent the risks of classifier over-fitting we use 75% recall cut-off for the Yahoo! web directory rare category classification. The basic over-sampling ends up with lowest average F1-Measure of, 76.09%. The variation of the F1-Measure with hierarchy depth, for 75% and 85% recall cut-off is shown in graph below.

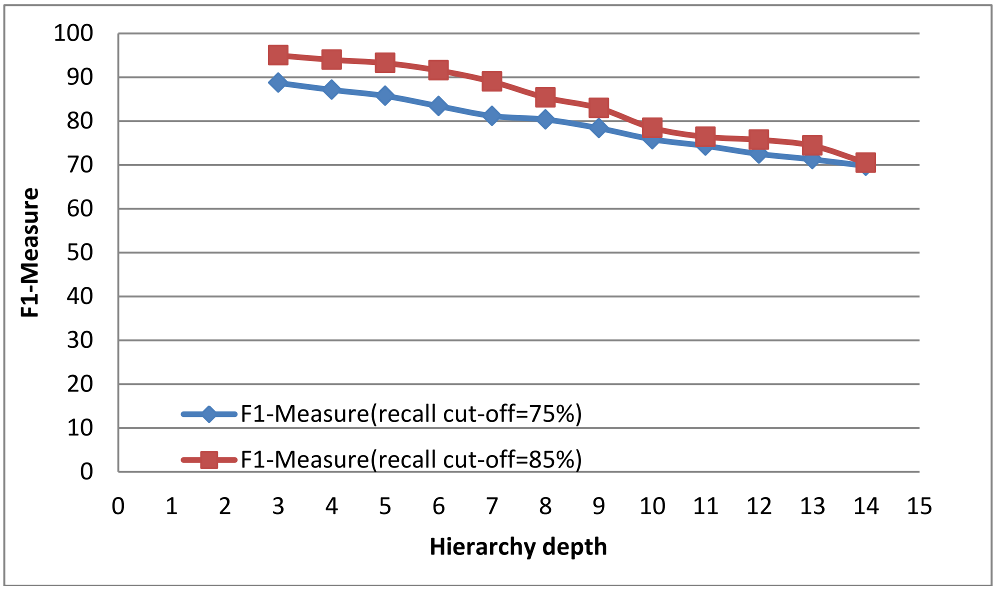

Figure 10: Average Rare Category Performance for Different Adaptive Over-Sampling Experiments

7.5 Results and Discussions

Ensemble learning and Incremental sampling based learning are the two popular large-dataset learning approaches. Combining these approaches with popular classifiers such as

Perceptron, one-class SVM, two-class SVM, and Maximum Entropy classifiers, a series of experiments using Yahoo! categories of more than 1000 positive instances have been conducted. In this research, compared to incremental sampling based learning, the ensemble architecture combined with a Maximum Entropy Classifier produced a much superior and cost effective performance.

In this research, the focused over-sampling and under-sampling of the negative instances is found effective to address the class imbalance associated with categories of 100 to 1000 web pages. Over-sampling is a popular data level approach to address rarity. Over-sampling in its basic form duplicates the rare category dataset. However, crude over-sampling may lead to classifier over-fitting. Advanced over-sampling of the rare category web pages, adaptive over-sampling, is found effective to address the rarity. The optimal solutions for class imbalance, rarity issue associated with categories of 10 to 100 web pages and large sample learning issues identified by this research are summarized in Table 16.

Table 16: Best Performing Classification Solutions for Imbalanced Dataset

Category Size	Machine Learning Issues	Best performing architecture	Average F1-Measure
1000 to 600,000	Class imbalance, over-abundance of positive instances	Ensemble architecture combined with maximum entropy classifier	87.86
100 to 1000	Class imbalance	Single maximum entropy classifier with focused over/under sampling of the negative instances	76.19
10 to 100	Rarity, class imbalance	Adaptively oversampled maximum entropy classifier	79.02

CHAPTER 8: AN OPTIMAL SOLUTION FOR CONTENT BASED LARGE-SCALE WEB PAGE CLASSIFICATION

8.1 Introduction

This research investigates a scalable and effective methodology to fulfill the classification requirements of popular web directories. In a hierarchical dataset similar to the Yahoo! Web directory, the prior probability distribution of subcategories indicates the presence or absence of class imbalance, rarity and the overabundance of positive instances within the dataset. The best performing classification techniques for these machine learning issues has been investigated in Chapters 4, 5, 6 and 7. Based on these results, in this chapter, we design a unified framework for the classification purpose of the complete Yahoo! Web directory.

The overall goals of this chapter are to address the following queries:

1. How can the automatic classification of the Yahoo! web directory be achieved and what is the best performance?
2. Are the arbitrarily fixed ranges for very-large categories, reasonably sized categories, rare categories, and absolute rare categories are justifiable?
3. What are the reasons for the improved classifier performance achieved in this research?

However, before discussing our Yahoo! Web directory classification model, we briefly review the spectrum of different hierachical classifier evaluation methods and also briefly disscuss about the hierchical classifier design and evaluation followed by this research.

8.2 Hierachical Classifier Evaluation:The Significance and Challenges

Over the years, classification research has emerged as a matured branch of machine learning with a large number of classification methods, each with different strengths and advantages. Enormous effort has been made to improve the basic classification methods but this has been only marginally succefull. With the large number of classification methods and results the consumers and researchers are equally uncertain about its worth, making future prospects of the classification research unclear.

The classification procedure used by the majority of researchers consists of selecting an evaluation metric, selecting a dataset, selecting a convincing number of previously designed strong learning algorithms to be compared to one another or compared against a new proposed method, and running stratified or random t-fold cross-validation experiments on the dataset domain. The average performance is calculated using popular evaluation metrics such as the F1-Measure, Precision/recall or accuracy.

While the machine learning community keeps busy with extensive comparisons of different classification algorithms in between or against a new proposed method, there is a strong need for empirical comparisons of different evaluation measures, to better identify their similarities and differences. Such empirical comparisons are not trivial due to the existing lack of clear definitions for the best evaluation criteria.

Based on the nature and structure of the dataset to be dealt with, classifiers can be evaluated in multiple ways. In a balanced flat dataset classification, an optimal evaluation of the methodology is mostly achieved by validating the system using a separate evaluation dataset. In this case, the average performance for each category after binary or multi-way classification is calculated and globally averaged into micro or macro measures.

According to Freitas et. al, (Freitas, A. A., & Carvalho, A. C. P. F., 2007) the problem of hierarchical classifier evaluation can be addressed using four different approaches. The first approach is known as "transformation of the hierarchical problem into a flat classification problem". In this approach, taking the leaf node classes alone, the hierarchical dataset classification is reduced to the flat classification problem. Later, flat dataset classification methods are applied.

The second approach is named as "hierarchical prediction with flat classification algorithms". In this approach, a hierarchical problem is divided into a set of flat classification problems. Typically one classifier is maintained for each level of the hierarchy. Later, the flat dataset classification methods are applied.

The third approach is known as "top-down approach". In this approach, one or more classifiers are trained for each level of the hierarchy. The root classifier is trained with all training examples. At the next class level, each classifier is trained with just a subset of the examples. The testing and evaluation starts with the root node. Based on the predictions produced by a classifier in each level, an example is classified in a top-down manner. Compared to the first two approaches, top-down approach produces a hierarchical classifier. However, in this approach, the errors made in higher levels of the hierarchy will be propagated to the most specific levels.

The forth approach is known as “big-bang approach”. In the big-bang approach, using the complete training set, a single iteration of the classification algorithm is performed to train the complete classification model. This approach increases the algorithmic complexity, but may avoid the risk of error propagation from the root.

The effectiveness of several other alternative methods to measure the predictive performance of a hierarchical classification algorithm including distance-based (Sun, A., & Lim, E. P., 2001), depth-dependent (Blockeel, H., Bruynooghe, M., Dzeroski, S., Ramon, J., & Struyf, J. , 2002), semantics-based (Sun & Lim, 2001) (Freitas, A. A., & Carvalho, A. C. P. F., 2007) and hierarchy-based measures (Ipeirotis, P) are also examined by the machine learning community. However, none of the discussed methods is frequently adopted by the machine learning community and there is not yet a consensus on which evaluation measure should be used in the evaluation of a hierarchical classifier. To our knowledge, no empirical comparisons of the discussed hierarchical classification evaluation methods have been reported. Such empirical comparisons are highly important to seek out the similarities and differences between the available classification evaluation methods. Such empirical comparisons are not trivial due to the existing lack of clear definition for the best evaluation criterion.

In addition to the aforementioned issues, most of the hierarchical classification problems need to address the instances with multiple labels. The effectiveness of the discussed hierarchical classifier evaluation measures when exposed to a dataset with multiple labels is not clear.

The validation of different evaluation schemes is quite time-consuming and not within the scope of this research. In the next section we will discuss the validation method we followed and our justifications for the same.

8.3 Content Based Classification of Yahoo! Web Directory

The prior probability distribution of a category indicates the presence or absence of relative rarity (class imbalance), alone or together with absolute rarity or large-sample learning issues due to the overabundance of positive instances. Based on the prior probability distribution and associated machine learning issues, we subdivided the subcategories of Yahoo! web directory into 5 mutually exclusive groups. These include very-large and imbalanced categories (categories of 1000 to 600,000 labeled instances), reasonably sized but imbalanced categories (categories of 100 to 1000 web pages), rare and imbalanced categories (categories of 10 to 100 labeled instances), extremely rare and imbalanced categories (categories of 1 to 10 web pages) and conceptual nodes. The effectiveness of different data level, algorithmic and architectural solutions to these machine learning issues is investigated in Chapters 4, 5, 6 and 7. These results are summarized as Table 17.

Table 17: A Comparison of Different Experiments Conducted on Yahoo! Web Directory

Category Size	Architecture	Average F1-Measure (%)
1000 to 600,000	Ensemble architecture combined with maximum entropy classifier	87.86
1000 to 600,000	Ensemble architecture combined with Perceptron Classifier (Information Gain feature selection)	66.07
1000 to 600,000	Ensemble architecture combined with Perceptron Classifier (Document Frequency feature selection)	85.26
1000 to 600,000	Ensemble architecture combined with One-Class SVM Classifier (Document Frequency feature selection)	70.61
1000 to 600,000	Ensemble architecture combined with Two-Class SVM Classifier (Document Frequency feature selection)	81.45
1000 to 600,000	Incremental sampling based learning combined with Maximum Entropy Classifier	79.54
100 to 1000	Single maximum entropy classifier	76.19
10 to 100	Adaptively oversampled maximum entropy classifier (recall cut-off 75%)	79.02
10 to 100	Adaptively oversampled maximum entropy classifier (recall cut-off 85%)	83.15
10 to 100	Crude over-sampling (recall cut-off 75%)	76.09

We designed a hierarchical machine learning model for the content based classification of the Yahoo! web directory. The best performing classification technologies for a particular prior probability distribution (Table 17) is used for this purpose. The model contains 132,342 classification units distributed in 14 layers. Each unit of this hierarchical classification model maps a Yahoo! web directory category of more than 10 labeled instances. Of the total 132,342 classification units, 988 units are ensembles of multiple member classifiers. The remaining classification units are modeled after sub-sampling or adaptive over-sampling. The first layer of the model contains 14 ensemble units only. The second layer of the model contains 196 classification units, of which, 133 units are ensemble classification model and remaining units are sub-sampled classification models designed for 63 reasonably sized and imbalanced categories. The distribution of ensemble, sub-sampled and adaptively over-sampled classification units in the designed Yahoo! classification model is summarized as Table 18.

We recursively assigned the contents of each category into its parent categories and designed a top-down classification model by integrating Ensemble Architecture, Sub sampled Architecture and Adaptively Over-Sampled Architecture trained on Maximum Entropy Model.

As mentioned earlier, the designed classification model has 14 layers and multiple classifiers are constructed at each level of the classification tree. Each classification unit within this hierarchy works as a flat binary classifier for a specific category. For each classification unit, the average F1-Measure after 3-fold cross validation is calculated. Then the Macro Averaged F1-Measure for each group of architecture at each level is calculated. Later, the average of the Macro-Average F1-Measure at each level, for each architecture, is separately calculated. The obtained value is reported as the average performance for the particular architecture. An unlabeled document will first be classified by the classifier at the root level into one or more lower level categories. It will then be further classified by the classifier(s) of the lower level categories until it reaches a final category which could be a leaf category or an internal category.

For an extremely imbalanced dataset similar to Yahoo! Web directory, the process of gathering a test set or evaluation set of the same prior probability distribution as the original dataset is statistically almost impossible. Hence we validated the model using a DMOZ subset.(Discussed as the Next Chapter)

With this classifier evaluation scheme, the duplicate instances due to the recursive expansion of the dataset do not mislead the evaluation process. However, global averaging of such results without taking the learning architecture used under consideration can be quiet misleading. This is because; each sub categories of a hierarchical dataset have different prior probability distribution and complexity of concept manifesting different combinations of the machine learning issues. Globally averaging such results will not reveal the impact of specific machine learning issue in the in the imbalanced dataset classification process.

With this classifier evaluation scheme, one may argue that the high average performance achieved by Yahoo! Classification model can be mostly contributed by a specific segment of the dataset say the upper level categories of the hierarchy or by a specific architecture say voted ensembles. We too noticed that the average performance is dropping with hierarchy depth and the average performance of the ensemble units are better compared to sub-sample or adaptively over sampled units. However, this research narrowed down the issues associated with large-scale web page classification research and the future research works in this area can focus more specific issues that we highlighted at the end of each chapter.

The average classifier performance across the hierarchy depth for different groups of Yahoo! categories is summarized in Table 19. The overall macro-average F1 Measure is 81.02.

This research is conducted on the Atlantic Computational Excellence Network (ACEnet). ACEnet is a High Performance Computing (HPC) environment providing distributed HPC resources, visualization and collaboration tools. Java 2 Enterprise Edition (J2EE) tools and architecture is used for software design, development and testing.

Table 18: The Structural Information of the Yahoo! classification model

level	**Adaptively over-sampled classification units**	**Sub-sampled classification units**	**Ensemble classification units**
1	0	0	14
2	0	63	133
3	995	728	233
4	8,420	2,222	347
5	11,530	2,895	138
6	28,705	2,161	87
7	33,997	1,082	33
8	16,569	529	3
9	10,981	229	
10	5,776	84	
11	3,043	22	
12	1,002	10	
13	268		
14	43		
Total	121,329	10,025	988

Table 19: Macro-Average F1 Measure Achieved for Yahoo! Web Directory Classification

Level	Categories with 1000+ labeled instances	Categories with 100 to 1000 labeled instances	Categories with 10 to 100 labeled instances
1	95.74		
2	95.01	91.33	
3	90.84	84.90	88.68
4	88.87	84.65	87.09
5	87.21	82.29	85.71
6	85.22	82.16	83.39
7	81.97	78.09	81.14
8	78.06	72.61	80.33
9		69.63	78.36
10		59.50	75.82
11		56.74	74.31
12			72.48
13			71.24
14			69.74
Average	87.86	76.19	79.02

8.3 Optimality of Arbitrarily Fixed Ranges for Machine Learning

This research fixed some arbitrary ranges for very large and rare categories. Whether these ranges are justifiable from the machine learning point of view is analyzed in this section.

While classifying very large datasets, the ensemble architecture is more promising than incremental learning. However, the definition for “large dataset” or number of positive instances required to form a “large dataset” is not clear. In this research, we defined all categories of more than 1000 labeled instances as large categories. In this section we analyze the appropriateness of this arbitrarily fixed range.

In our ensemble architecture, sub-sampling is performed on the original dataset to create multiple samples. Member classifiers are trained from each sample. Thus, an ensemble classifier comprises a group of member classifiers where each classifier maps the knowledge contained in the small segment of the original dataset. The category of a new web page is determined by voting by the member classifiers. If sufficient training instances have been taken, the ensemble formed by the multiple member classifiers does not result in any performance drop due to the sub-sampling. But the optimal sample size varies with the dataset. This makes sample size optimization for member classifier critical.

The sample size optimization followed in this research includes training multiple member classifiers taking increasingly larger samples, observing the trends, and stopping when no progress has been made. The sample size used in the first iteration and the increase in the sample size in the proceeding iterations are ensured large enough to be representative of the original dataset. This prevents the situation of two consecutive iterations not bringing any significant difference in the classifier performance due to an insufficient number of training instances, leading to wrong interpretations on the average performance. This is achieved by a set of predefined conditions to be followed while performing the sub-sampling. The summary of these conditions is as follows:

While classifying a very large category, say C_i of collective positive instances more than 100,000 instances, in the 1st, 2nd, 3rd, 4th, 5th, 6th, 7th, 8th and 9th iterations 2%, 4%, 6.25%, 8.33%, 10%, 20%, 33.33%, 50% and 100% of the labeled instances of C_i and all child categories are drawn and combined with an equal numbers of negative instances from the sibling categories of C_i. These iterations progress to 50, 25, 18, 12, 10, 5, 3, 2 and 1 unique samples of C_i and are created for member classifier design. Whereas, while classifying categories of 10,000 to 100,000 collective positive instances, the 1st iterations starts with 6.25% of the labeled instances and for categories of 1000 to 10,000 positive instances, in the 1st iteration starts with 10% of the positive instances from C_i and all child categories and an equal numbers of negative instances from the sibling categories of C_i. Thus we ensure an optimal sample size and an optimal step size across the categories of varying size.

In this architecture, the average performance of each iteration is evaluated using TP Rate, FP Rate and variance across TP and FP Rates. This prevents the situation of selecting a wrong group of member classifiers for ensemble learning with high average performance but high a degree of dissimilarity between them. Also, the average sample size is normalized to unity before calculating the slope between two consecutive sample size and QF. This prevents bias due to the minute fluctuations in the average sample size if any.

Moreover, the ensemble architecture converges to a single classifier if sub-sampling and ensemble learning is not profitable for the given category. This prevents the chances for data fragmentation and associated performance drop.

Taking the best performing Perceptron classifiers and Maximum Entropy classifiers combined with ensemble architecture, the percentage of categories converged into a single classifier is 3.03% and 2.43% respectively. This covers 93% and 89.56% of the categories of less than 1700 positive instances. For the remaining categories, the ensemble architecture converged between 6.25% and 50% of positive instances. Thus, in the given situation there is no evidence of data fragmentation associated with the lower range. The sub-sampling statistics for the Perceptron and Maximum Entropy classifier is as shown in Figure 11 and 12.

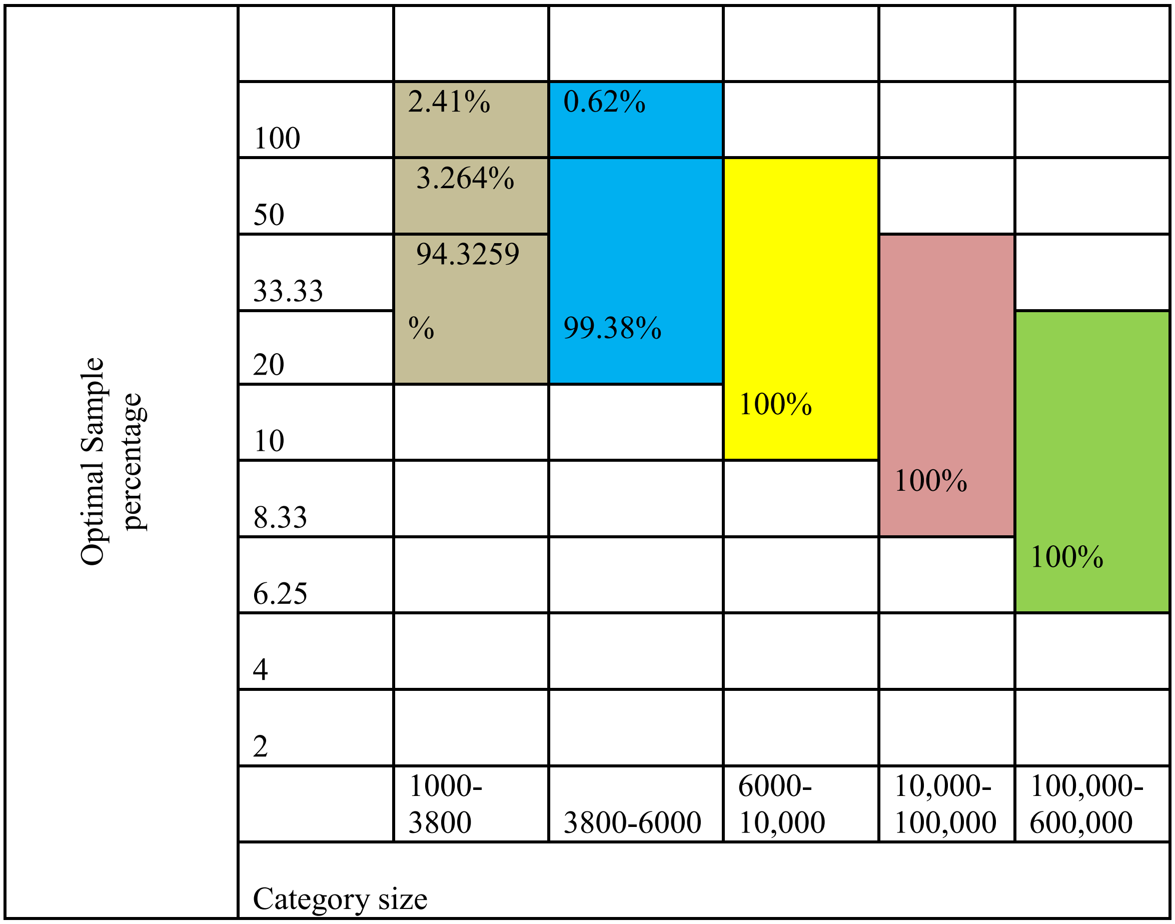

Figure 11:Sub-Sampling Statistics for Ensemble Learning Combined With Perceptron Classifiers and Document Frequency Feature Selection Method

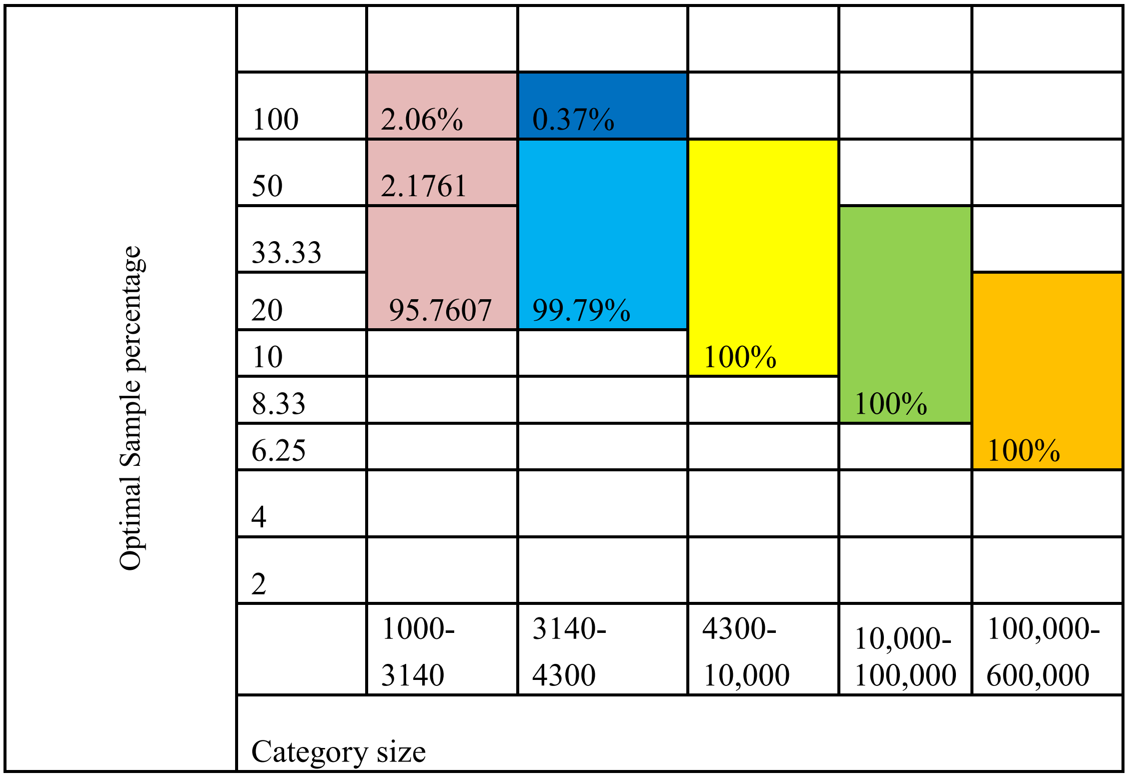

Figure 12: Sub-Sampling Statistics for Ensemble Learning Combined With Maximum Entropy Classifiers

Similarly, the sample size that makes "rarity" is also not predefined and is highly domain dependent. The optimality of the fixed rage (10 to 100 positive instances) and recall cut-off of 75% for rare categories is analyzed. For the categories of 10 web pages, the highest percentage of oversampling is 70%. If the categories of 9 and 8 labeled instances were included, the percentage of over-sampling could have been 90% and 100% respectively. Thus, the probability for the number of categories with exact duplicate samples resulting in classifier over-fitting will be higher if we would have been fixed a lower limit of less than 10 positive instances for rare categories. Similarly, an achieved benefit of 4.83% in the average F1-Measure, associated with 16.07% increase in the average over-sampling for a recall cut-off of 85%, is questionable. Hence, for Yahoo! rare categories, a recall cut-off of 75% is more suitable for adaptive over-sampling based learning.

While considering Yahoo! categories of 100 to 1000 labeled instances focused over sampling and undersampling has been performed, to address the class imbalance and associated issues. In this group, 5% of the categories are over-sampled. This includes over-sampling of negative instances. The highest percentage of over-sampling is 18.17%. For the remaining categories, the negative instances have been undersampled. The average performance of Yahoo! categories of this group at different category sizes is listed as Table 20 and compared with the average performance of categories with 10 to 100 web pages.

Table 20: Average F1-Measure of Yahoo! Categories With 100 to 1000 Labeled Instances

Category Size	Macro-F1 Measure
Categories of 100 to 500 positive instances	70.78
Categories of 500 to 1000 positive instances	81.6
Categories of 100 to 1000 positive instances	76.19
Categories of 10 to 100 positive instances(discussed in section:5.3)	79.02

The average F1-Measure of adaptivly over-sampled rare categories with 10 to 100 web pages, discussed in the next section, is 79.02%, which is 2.83% higher than the average F1-Measure of categories with 100 to 1000 web pages. However, the average F1-Measure of categories with 500 to 1000 is 81.6%, which is 2.58% higher than the average F1-Measure of categories with 100 to 1000 web pages. Thus, we could conclude that, with the current architecture, the categories with 100 to 500 web pages show a slight drop in the average classifier performance. However, all categories of more than 75 web pages is oversampled less than 30%, where as all categories of less than 45 Web pages is oversampled by more than 50%. The highest over-sampling is performed with the categories of less than 20 labeled web pages. Based on the results of rare category classification discussed in the next section, a slight improvement on the classification of categories with 100 to 500 web pages can be achieved by adaptive over-sampling.

8.4 The Reasons for Improved Average Performance Achieved in this Research

The class imbalance, rarity and large-sample learning issues within web directories make applying feature reduction techniques and classification algorithms very difficult. The earlier large scale web page classification research works either overlooked the machine learning issues due to rarity, class imbalance and over-abundance of training instances or addressed these issues using a common framework. These researches either lead to highly comprehensive or inadequate statements such as traditional web page classification techniques are insufficient to address the challenges of large-scale web page classification problem, or provide limited insight to the real challenges of large-scale web page classification.

In our opinion, addressing class imbalance, rarity and overabundance of positive instances within a dataset using a generalized methodology is highly insufficient. Similarly, the class imbalance can be either positive dominant or negative dominant and should be addressed separately using appropriate architecture or algorithm. We designed a unified framework applicable for the classification of imbalanced dataset of any size with extreme rarity. In this research, based on the prior probability distribution and associated machine learning issues, we subdivided the entire Yahoo! categories into 5 mutually exclusive groups. The effectiveness of different data level, algorithmic and architectural solutions to these machine learning issues is explored. The best performing classification technologies for a particular prior probability distribution has been identified and integrated to the Yahoo! web directory classification model. Later the methodology used for Yahoo! categorization is evaluated using a DMOZ subset and we statistically proved that the methodology works equally well when applied to any content based hierarchical web page classification of larger or smaller dataset.

CHAPTER 9: VALIDATION OF THE METHODOLOGY USING DMOZ SUBSET

9.1 Introduction

The effectiveness of different data level, algorithmic and architectural solutions to the over-abundance of positive instances, class imbalance and rarity problems associated with large-scale web page classification is examined in this research. These methods, combined with the Perceptron, Support Vector Machine and Maximum Entropy Classifiers, have been analyzed and the best performing classification technologies have been applied to classify the complete Yahoo! web directory of 639,671 categories and 4,140,629 web pages. Whether the methodology of this research will work equally well when applied to the content based hierarchical web page classification of larger or smaller dataset is examined in this chapter. These experiments are conducted on a hierarchical subset of the DMOZ web directory.

9.2 Categorization of DMOZ Subset

At the time of our crawling in October, 2009, there were 602,410 categories and 4,519,050 web pages in the topmost 14 levels of the DMOZ web directory. The category distribution of the DMOZ web directory with hierarchy depth is similar to that of Yahoo! web directory and is as shown in Figure 13. Like the Yahoo! web directory, most of the DMOZ categories are extremely rare with fewer than 10 labeled web pages.

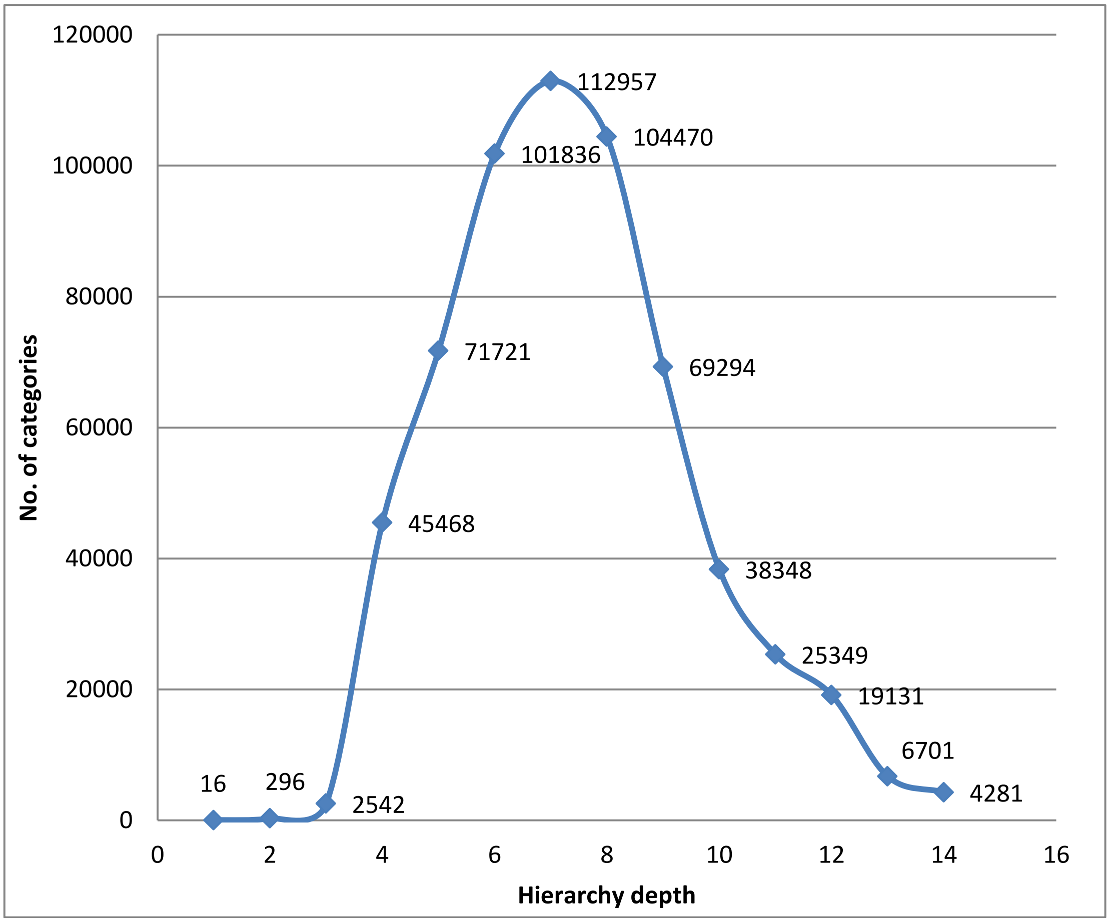

Figure 13: DMOZ Web Directory Category Distribution with Hierarchy Depth

The effectiveness of the best performing Yahoo! Web directory classification technologies are analyzed using a DMOZ subset of 17,217 categories and 130,594 web pages. This dataset is downloaded from the Large Scale Hierarchical Text classification (LSHTC) Pascal Challenge (http://lshtc.iit.demokritos.gr/). The LSHTC Challenge is a hierarchical text classification competition using large datasets based on DMOZ web directory. The category distribution of this subset with hierarchy depth is as shown in Figure 9. The detailed category distribution of this subset is summarized in Table 21.

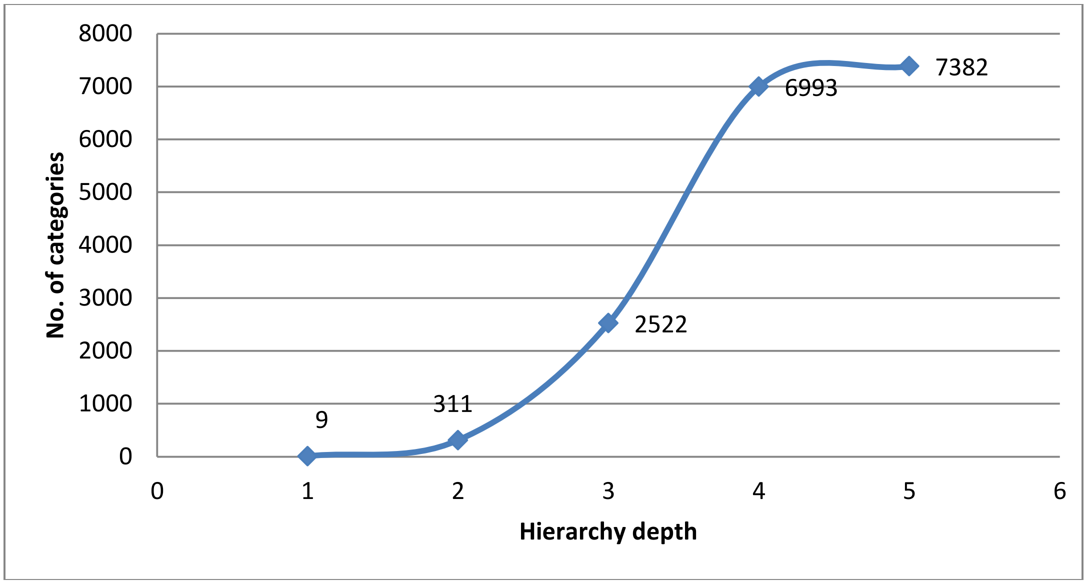

Figure 14: Category Distribution of DMOZ Subset

We partitioned the categories of the DMOZ subset into five mutually exclusive groups as shown in Table 22. There are 62 DMOZ categories containing 1000 to 100,000 web pages. Ensemble learning combined with the Maximum Entropy Classifiers, the best performing classification technology when applied to the Yahoo! categories of more than 1000 positive instances, is applied to classify the DMOZ categories of this group.

Table 21: Detailed category distribution of DMOZ subset

Hierarchy depth	Total Categories	Categories with 1000+ labeled web pages	Categories with 100 to 1000 web pages	Categories with 10 to 100 web pages	Categories with 1 to 10 web pages
1	9	9	0	0	0
2	311	33	112	118	48
3	2522	15	242	1144	1121
4	6993	4	175	2276	4538
5	7382	1	103	2111	5167

Table 22: DMOZ Subset Category Distribution

Group	Number of categories
Categories with more than 1000 labeled web pages	62
Categories with 100 to 1000 labeled web pages	632
Categories with 10 to 100 web pages	5,649
Categories with 1 to 10 web pages	10,873
Categories without any labeled web pages	0

There are 632 DMOZ categories containing 100 to 999 web pages. These are reasonably sized categories for efficient machine learning; however, dominance or scarcity of negative instances of the sibling categories results in the class imbalance.

In addition to the class imbalance, 5,649 categories of this DMOZ subset are rare categories of 10 to 100 positive training instances. Adaptive over-sampling combined with Maximum Entropy Classifiers, the best performing categorization technique for Yahoo! categories of 10 to 100 web pages, is used to classify these categories. There are 10,873 Yahoo! categories containing 1 to 9 web pages. Due to the lack of training instances, no individual classifiers have been designed for this group.

The macro-averaged F1-Measure of DMOZ subset achieved in this research is 84.85%. The highest average F1-Measure reported for this dataset in LSHTC Pascal Challenge is 35.49% (http://lshtc.iit.demokritos.gr/node/23). In their research, whether any hierarchy pruning or expansion has been performed prior to the classification is not clear. The average F1-Measure and rare category recall for categories with more than 1000 web pages achieved for DMOZ subset is 86.27% and 79.68% respectively. Average F1-

Measure and rare category recall for categories with 100 to 1000 web pages is 84.3% and 77.50% respectively. For categories with 10 to 100 web pages, the achieved average F1-Measure and rare category recall is 83.99% and 77.86% respectively. Adaptive over-sampling with 75% recall cut-off is used to set up this experiment. The average classifier performance of DMOZ subset is summarized in Table 23.

Table 23: Average Classifier Performance of DMOZ Subset

Category Group	# of categories	Methodology	Macro-F1	Rare category recall
1000-600,000 labeled web pages	62	Ensemble architecture combined with maximum entropy classifier	86.27	79.68
100-999 labeled web pages	632	Single maximum entropy classifier	84.3	77.5
10-99 labeled web pages	5649	Adaptively oversampled maximum entropy classifier	83.99	77.86
1-9 labeled web pages	10,873	Macro averaged Recall :70.58		

To check the significant difference in the average F1-Measures between Yahoo! web directory and DMOZ subset if any, a two-variable Chi-Square test of independence is conducted. The Macro-averaged-F1 Measures of the Yahoo! and DMOZ subset is used for this. The two variable chi-square test of Independence is discussed as the next section.

9.3 Test of Independence

The two variables considered for test of independence using Chi-square test are F1-Measure of the 3 groups of Yahoo! and DMOZ subset categories (categories with more than 1000 labeled instances, categories with 100 to 1000 labeled instances and categories with 10 to 100 labeled instances). The formula for Chi-square test of independence is

$X^2=\sum (O-E)^2 / E$, where O is the observed frequency, and E is the expected frequency. The degree of freedom for the two-dimensional chi-square statistic is defined as,

df = (C - 1) x (R - 1) where C is the number of columns or levels of the first variable and R is the number of rows or levels of the second variable. In our case, C is the number of datasets, which are 2. R is the three sets of Macro-averaged F1-Measure for each dataset. Hence the degree of freedom, (3-1) x (2-1), is 2. The five step process for testing statistical hypotheses for our research problem is as follows:

*1. State the null hypothesis and the alternative hypothesis based on the research question***.**

H_0: The methodology of this research works equally well on small datasets.

H_1: The methodology of this research is not applicable to small datasets.

*2. Set the alpha level***.**

We set ∞ as 0.025.

3. Calculate sum of Chi-square value (Table 24)

Table 24: Chi-Square Test Result on the Macro-Averaged F1-Measure of Yahoo! Web Directory and DMOZ Subset

Dataset	F1-Measure	chi-square value
Yahoo!: categories of 1000+ labeled instances	87.86	0.292062062
Yahoo!: categories of 100 to 1000 labeled instances	76.19	0.549077482
Yahoo!: categories of 10 to 100 labeled instances	79.02	0.185114415
DMOZ: categories of 1000+ labeled instances	86.27	0.13383714
DMOZ: categories of 100 to 1000 labeled instances	83.99	0.013336093
DMOZ: categories of 10 to 100 labeled instances	84.3	0.022356702
Total		1.9588

4. Write the decision rule for accepting the null hypothesis

The critical value for Chi-square for 0.025 level and degree of freedom 2 is 7.378. Hence, null hypothesis H_0 will be accepted if sum of chi-square value is less than 7.378.

5. Conclusion

The obtained value of Chi-square for Macro-averaged F1-Measure of DMOZ and Yahoo! subset is 1.9588. Hence the null hypothesis is accepted. We conclude that the methodology works equally well on large and small datasets.

9.4 Conclusions

We designed a unified framework applicable for the content based classification of any imbalanced dataset with extreme rarity. Methodology of this framework is tested on the complete Yahoo! web directory of 639,671 categories and 4,140,629 web pages. Later, the methodology is evaluated using a DMOZ subset of 17,217 categories and 130,594 web pages and we statistically proved that the methodology used works equally well on large and small dataset. A comparison of the average performance of complete Yahoo! web directory and DMOZ subset is summarized as Table 25.

Table 25: A Comparison of Yahoo! Web Directory and DMOZ Subset Classification

Category Size	**Yahoo!**		**DMOZ**	
	F1-Measure	**Rare Category Recall**	**F1-Measure**	**Rare Category Recall**
1000+ labeled web pages	87.86	80.44	86.27	79.68
100 to 1000 labeled web pages	76.19	67.49	83.99	77.86
10 to 100 labeled web pages	79.02	65.34	84.3	77.5
1 to 10 labeled web pages	Macro averaged Recall:71.09		Macro averaged Recall:78.34	

Comparison of our results with other large-scale web page classification is summarized in Table 26. This table presents the cardinal comparison only as the hierarchical classifier evaluation procedure other researchers followed to calculate the reported F1 measure is not clear.

Table 26: A Comparison of Our Results with Other Large-Scale Web Page Classification Research

Researcher	Dataset	No of Categori es	Dept h	Method	Micro-F1 (%)	Macro -F1 (%)
Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., & Ma, W., 2004	Yahoo!	246,279	14	Hierarchical SVM	24	12
Chen, 2000; Dumais, S., & Chen, H., 2000	LookSmar t	163	2	Hierarchical SVM		52.4
Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008	ODP	130,000	17	Statistical language model	51.8 (at 5th level)	
Xue, G., Xing, D., Yang, Q., & Yu, Y., 2008	ODP	130,000	17	Hierarchical SVM	29.2 (at 5th level)	
Marath, S., Shepherd, M., Duffy, J., Milios, E., & Heywood, M.,2009	Yahoo!	639,671	17	Maximum Entropy Classifier		81.02
Marath, S., Shepherd, M., Duffy, J., Milios, E., & Heywood, M., 2009	DMOZ subset	17,217	5	Maximum Entropy Classifier		84.85
Jhuang, LSHTC Challenge	DMOZ subset	17,217	5	NA		35.49

CHAPTER 10: CONCLUSION AND FUTURE RESEARCH

Traditional classification algorithms assume the target classes of the dataset share similar prior probability distribution. However, in real-world datasets like web taxonomies, and intrusion dataset this identical prior probability assumption is violated. For example, the popular web directories have hundreds of thousands of categories, deep hierarchies, class imbalance and rarity (very small classes) within the dataset. These properties make applying classification algorithms to such data sets very difficult. The available classification results on reasonably sized subsets of popular web directories conclude that in terms of effectiveness, the popular classification algorithms cannot fulfill the classification needs of very large-scale taxonomies.

We investigated a scalable and effective methodology to fulfill the classification requirements of popular web directories. To start with, we conducted the statistical analysis of Yahoo! and DMOZ web directories. In Yahoo! web directory, 0.16% of the total categories (988 categories) are very large, containing 1000 to 600,000 labeled web pages of their own; whereas, 19.06% of the Yahoo! categories are absolutely rare categories of 10 to 100 labeled web pages. Classification algorithms, when applied to very large categories of more than 1000 labeled instances should address the machine learning issues due to the class imbalance and large-sample learning. Conversely, classification algorithms, when applied to rare categories of 10 to 100 labeled web pages should address the machine learning issues due to class imbalance and rarity. Another 1.58% of Yahoo! categories are reasonably sized categories holding 100 to 1000 labeled web pages. However, the abundance or shortage of negative instances in the sibling categories makes these categories imbalanced. There are 504,240 Yahoo! categories containing 1 to 9 web pages. This forms 78.82% of total Yahoo! categories.

The earlier large scale web page classification research works either overlooked the machine learning issues due to rarity, class imbalance and over-abundance of training instances or addressed these issues using a common framework. These researches either lead to highly comprehensive or inadequate statements. Therefore they are insufficient to address the challenges of large-scale web page classification problem, or provide limited insight to the real challenges of large-scale web page classification.

In our opinion, addressing multiple machine learning issues within in a dataset of hundreds of thousands of categories issues using a generalized methodology is highly insufficient. In this research we analyzed the prior probability distribution of each Yahoo! subcategory and applied appropriate classification techniques. Later the methodology used for Yahoo! categorization is evaluated using a DMOZ subset and we statistically proved that the methodology works equally well when applied to any content based hierarchical web page classification of larger or smaller dataset.

There are a few areas in large-scale web page classification that need more investigation. The impact of class imbalance on the popular feature selection measures is an unexplored area of this research. However, preliminary studies are conducted and we conclude that statistical feature selection methods such as Information Gain are not optimal for the classification of very large web directories.

At this point, extreme rarity prevents training individual classifiers for categories with fewer than 10 labeled web pages. We cannot expect any statistical learner to perform well on such rare categories. In this research, the classifiers of the parent categories have been used to classify these categories. The advantage of merging extreme rare categories to the parent categories is applicable to the hierarchical dataset only. Around 70% of the categories of the popular web directories are extremely rare with fewer than 10 labeled instances. A better alternative to categorize these categories will complement many real-world flat and hierarchical classification problems including text classification, medical dataset classification, intrusion detection, etc., where extreme rarity is an inexhaustible challenge.

REFERENCES

Abe, N., Zadrozny, B., & Langford,J. (2004). An Iterative Method for Multi-Class Cost-Sensitive Learning. *Proc. ACM SIGKDD Int'l*, (pp. 3-11).

Akbani,R., Kwek ,S., & Japkowicz,N. (2004). Applying support vector machines to imbalanced datasets. *15th European Conference on Machine Learning (ECML)*, (pp. 39-50).

Apte, C., Damerau, F., & Weiss, S. M. (1994). Towards language independent automated learning of text categorization models. *Proceedings of the 17th Annual ACM/SIGIR Conference.*

Barandela,R.,Sánchez,J.S.,García,B.V., & Rangel,E. (2003). Strategies for learning in class imbalance problems. *Pattern Recognition* , 849-851.

Blockeel, H., Bruynooghe, M., Dzeroski, S., Ramon, J., & Struyf, J. . (2002). Hierarchical multi-classification. *In Proceedings of the ACM SIGKDD 2002 Workshop on Multi-Relational Data Mining (MRDM 2002).*, (pp. 21-35).

Blum. A., & Mitchell. T. (1998). Combining labeled and unlabeled data with co-training. *Proceedings of the 11th Annual Conference on Computational Learning Theory.* Morgan Kaufmann Publishers.

Breiman, L. (1996). Bagging Predictors. *MachineLearning* , 123-140.

Catlett, J. (1991). Mega-induction:a test flight. Proceedings of the eighth International Conference on Machine Learning(ICML). Michigan.

Chakrabati. S., Dom. B.,& Indyk. P. (1998). Enhanced hypertext categorization using hyperlinks. *ACM International conference on management of data(SIGMOD)*, (pp. 307-318).

Chawla, N.V.,Bowyer,K.W.,Hall,L.O., and W. P. Kegelmeyer. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 321-357.

Chen, K. L. (2006). Efficient Classification of Multi- Label and Imbalanced Data Using Min-Max Modular Classifiers. *Proc. World Congress on Computation Intelligence—Int'l Joint Conf.Neural Networks*, (pp. 1770-1775).

Chen, H. (2000). Bringing order to the web:automatically categorizing search results. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* , 145-152.

Cohen W., & Singer Y. (1999). Context-sensitive learning metods for text categorization. *ACM Transactions on Information Systems* , 141-173.

Creecy, & Robert, H. (1992). Trading MIPS and memory for knowledge engineering:Classifying census returns on the connection machine. *Communicaion of the ACM* , 48–63.

Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. In *Multiple Classifier Systems* (pp. 1-15). Springer Berlin / Heidelbe$^{rg.}$

Doucette, J. & Heywood,M.I. (2008). GP Classification under Imbalanced Data Sets: Active Sub-Sampling AUC Approximation. *Lecture Notes in Computer Science*, pp. 266-277.

Drummond, C., & Holte, R. C. (2003). C4.5, Class Imbalance, and Cost Sensitivity: Why Under-sampling beats Over-sampling. *In Workshop on Learning from Imbalanced Data Sets* .

Dumais, S.,& Chen, H. (2000). Hierarchical classification of Web content. *SIGIR*, (pp. 256-263).

Ertekin, S., Huang, J., & Giles, C.L. (2007). Active Learning for Class Imbalance Problem. *Proc. Int'l SIGIR Conf. Research and Development in Information Retrieval*, (pp. 823-824).

Ertekin, S., Huang,J., Bottou,L., & Giles, L. (2007). Learning on the Border: Active Learning in Imbalanced Data Classification. *Proc. ACM Conf. Information and Knowledge Management*, (pp. 127-136).

Freund, Y., & Schapire,E. (1999). A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence* , 771-780.

Freitas, A. A., & Carvalho, A. C. P. F. (2007). A Tutorial on Hierarchical Classification with Applications in Bioinformatics. Research and Trends in Data Mining Technologies and Applications, 176-209.

Friedman, J.H., Kohavi, R., & Yun, Y. (1996). Lazy decision trees. *Thirteenth National Conference on Artificial Intelligence*, (pp. 717-724).

Fuhr, N., Hartmanna, S., Lustig, G., Schwantner, M., & Tzeras, K. (1991). Air/x—A rule-based multistage indexing systems for large subject fields. *Proceedings of RIAO*, (pp. 606–623).

Furnkranz, J. (1999). Exploiting structural information for text classification on the WWW. In *Advances in Intelligent Data Analysis* (pp. 487–497). Springer Berlin / Heidelberg.

Glover,E.J., Tsioutsiouliklis,K., Lawrence,S., Pennock, D.M., Flake, G.W. (2002). Using web structure for classifying and describing web pages. *Proceedings of international World Wide Web conference*, (pp. 562–569).

Harris-Jones, C., & Haines, T.L. (1997). *Sample Size and Misclassification:Is More Always Better.* AMS Center for Advanced Technologies.

He, H., Bai,Y.,Garcia,E.A., & Li,S. (2008). ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. *Proc. Int'l J. Conf. Neural Networks*, (pp. 1322-1328).

Holte,R.C., Acker, L.E., & Porter,B.W. (1989). Concept learning and the problem of small disjuncts. . *In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, (pp. 813-818).

Huang, K.Z., Yang,H.Q., King, I.,& Lyu, M.R. (2004). Learning Classifiers from Imbalanced Data Based on Biased max Probability Machine. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*

Ipeirotis, P., Gravano, L., & Sahami, M. . (2001). Probe,count, and classify: categorizing hidden web databases. Proceedings of the 2001 ACM SIGMOD international conference on Management of data., (pp. 67-78).

Japkowicz,N.,& Stephen,S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis* , 203-231.

Japkowicz, N. (2002). Supervised learning with unsupervised output separation. *Proc. ASC 2002*, (pp. 321-325).

John, M.P. (2000). Practical issues for automated categorization of webpages. *ECDL 2000 workshop on the semantic web.*

John. G., Kohavi, R., Pfleger, K. (1994). Irrelevant features and the subset selection problem. *Machine learning: proceedings of the 11th international conference.*, (pp. 121–129).

Kang, P. & Cho,S. (2006). EUS SVMs: Ensemble of Under sampled SVMs for Data Imbalance Problems. *Lecture Notes in Computer Science*, pp. 837-846.

Kohavi, R.,& John,G.H. (1997). Wrappers for feature subset selection. *Artificial Intelligence Journal* , 273.

Kotsiantis,S., Kanellopoulos, D.,& Pintelas,P. (2006). Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering* , 25-36.

Lee, H.,J., & Cho, S. (2006). The Novelty Detection Approach for Difference Degrees of Class Imbalance. *Lecture Notes in Computer Science*, pp. 21-30.

Lewis, D. D., & Ringuette, M. (1994). Comparison of two learning algorithms for text categorization. *Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval.*

Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). RCV1: a new benchmark collection for text categorization research. *Journal of Machine Learning Research* , 361-397.

Lewis, D.D., Schapire, R.E., Callan,J.P., Papka, R. (1996). Training algorithms for linear text classifiers. *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 298-306).

Lin, Chang,C. & Jen,C. (2001). *LIBSVM: a library for support vector machines.* Retrieved 2008, from http://www.csie.ntu.edu.tw/~cjlin/libsvm

Ling, C.& Li,C. (1998). Data mining for direct marketing problems and solutions. *In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, (pp. 73-79).

List of stop words, Information Retrieval Resources. (n.d.). Retrieved 2008, from http://nlp.stanford.edu/IR-book/information-retrieval.html

Liu,T., Yang, Y., Wan,H., Zeng,H., Chen,Z.,& Ma,W. (2004). Support Vector Machines Classification with A Very Large-scale Taxonomy. *SIGKDD Explorations* , 1-36.

Liu, X.,Y., & Zhou, Z., H. (2006). Training Cost-Sensitive Neural Networks. *IEEE Trans. Knowledge and Data Eng.*, 63-77.

Liu, X.Y. & Zhou, Z.H. (2006). Training Cost-Sensitive Neural Networks with Methods Addressing the Class Imbalance Problem. *IEEE Trans. Knowledge and Data Eng.*, 63-77.

Liu, X.Y., Wu,J., & Zhou.Z.H. (2009). Exploratory under-sampling for class imbalance learning. *IEEE Transactions on System,Man, and Cybernetics-Part B Cybernetics*, 539-550.

Liu, Y., An, A., & Huang, X. (2006). Boosting Prediction Accuracy on Imbalanced Data Sets with SVM Ensembles. *Lecture Notes in Artificial Intelligence.*, pp. 107-118.

Liu., X.Y & Zhou.,Z.H. (2006). The Influence of Class Imbalance on Cost-Sensitive Learning: An Empirical Study., (pp. 970-974).

MEGA Model Optimization Package. (2007). Retrieved 2008, from , http://www.cs.utah.edu/~hal/megam/

Manevitz, L., & Yousef, M. (2007). One-Class Document Classification via Neural Networks. *Neurocomputing*, pp. 1466-1481.

McCarthy, K.,Zabar,B., & Weiss,G.M. (2005). Does Cost-Sensitive Learning Beat Sampling for Classifying Rare Classes? *Proc. Int'l Workshop Utility-Based Data Mining*, (pp. 69-77).

Mease, D., Wyner,A.J., & Buja,A. (2007). Boosted Classification Trees and Class Probability/Quantile Estimation. *Machine Learning Research*, 409-439.

Mladenic. D., & Grobelnik. M. (1999). Feature selection for unbalanced class distribution and Naive Bayes. *16th International conference on machine learning*, (pp. 258–267).

Mladenic. D.,& Grobelnik. M. (1998). Feature selection for classification based on text hierarchy: Working notes of learning from text and the web. *Conference on automated learning and discovery.*

Monard,M.C.,& Batista,G . (2002). Learning with Skewed Class Distribution. *Advances in Logic, Artificial Intelligence and Robotics* , 173-180.

Moulinier, I., Raskinis, G., & Ganascia, J. (1996). Text categorization: A symbolic approach. *Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval.*

Ng, H.T., Goh, W.B.,& Low, K.L. (1997). Feature selection, perceptron learning, and a usability case study for text categorization. *ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 67-73).

Oh, H., Myaeng, S.H.,& Lee, M. (2000). A practical hypertext categorization method using links and incrementally available class information. *ACM international conference on research and development in information retrieval*, (pp. 264–271).

Provost, F.,& Fawcett,T. (2001). Robust classification for imprecise environment. *Machine Learning* , 203-231.

Raskutti, B.& Kowalczyk,A. (2004). Extreme rebalancing for svms: a case study. *SIGKDD Explorations* , 60-69.

Sebastiani, F. (1999). A tutorial on automated text categorization. *Proceedings of of ASAI-99: 1st Argentinean symposium on artificial intelligence*, (pp. 7–35).

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys* , 1-47.

Seiffert, C. Khoshgoftaar,T.M., Van Hulse,J., & Napolitano,A. (2007). Mining data with rare events:A case study. *ICTAI*, (pp. 132-139).

Su, C.T., & Hsiao,Y.H. (2007). An Evaluation of the Robustness of MTS for Imbalanced Data. *IEEE Trans. Knowledge and Data Eng.*, pp. 1321-1332.

Sun, A., & Lim, E. P. (2001). Hierarchical text classification and evaluation. In Proceedings of the 2001 IEEE International Conference on Data Mining., (pp. 521–528).

Sun, Y., Kamel, M.S., & Wang,Y. (2006). Boosting for Learning Multiple Classes with Imbalanced Class Distribution. *Proc. Int'l Conf. DataMining*, (pp. 592-602).

Tang, Y., & Zhang,Y.,Q. (2006). Granular SVM with Repetitive Undersampling for Highly Imbalanced Protein Homology Prediction., (p. Proc. Int'l Conf. Granular Computing).

The Porter Stemming Algorithm. (2006). Retrieved 2007, from http://tartarus.org/~martin/PorterStemmer/

Tikk,D., Bansaghi,Z.,& Biro,G. (2005). Categorizing Gigabytes: Experiments on the RCV1 Corpus. *6th Int. Symp. of Hungarian Researchers on Computational Intelligence*, (pp. 276-276).

Tzeras, K.,& Hartman, S. (1993). Automatic indexing based on bayesian inference networks. *Proc. 16th Ann. Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 22-34).

Veropoulos,K., Campbell,C., & Cristianini,N. (1999). Controlling the sensitivity of support vector machines. *Proceedings of the International Joint Conference on AI*, (pp. 55-60).

Vilarino, F., Spyridonos,P., Radeva,P., & Vitria,J. (2005). Experiments with SVM and Stratified Sampling with an Imbalanced Problem: Detection of Intestinal Contractions. *Lecture Notes in Computer Science*, pp. 783-791.

Visa, S., & Ralescu,A. (2005). Issues in Mining Imbalanced Data Sets-A Review Paper. *Proceedings of the Sixteen Midwest Artificial Intelligence and Cognitive Science Conference* , 67-73.

Wang, B.,X., & Japkowicz,N. (2008). Boosting Support Vector Machines for Imbalanced Data Sets. *Lecture Notes in Artificial Intelligence*, pp. 38-47.

Weiss, G. M. (2004). Mining with Rarity: A Unifying Framework. *Sigkdd Explorations*, 7.

Weiss,G.M.,& Hirsh,H. (2000). A quantitative study of small disjuncts. *In Proceedings of the Seventeenth National Conference on Artificial Intelligence*, (pp. 665-670).

Weiss,G.M.,& Provost,F. (2003). Learning when training data are costly:the effect of class distribution on tree induction. *Artificial Intelligence Research* , 315-354.

Weiss.G.M, (2003). *The effect of small disjuncts and class distribution on decision tree learning.* PhD Dissertation,Department of Computer Science,Rutgers University,New Jersey.

Weiss,S.M., Apte, C., Damerau, F.J., Johnson, D.E., Oles, F.J., Goetz, T., Hampp, T. (1999). Maximizing text-mining performance. *Intelligent Systems and Their Applications* , 63 – 69.

Weiss,S.M.,& Indurkhya,N. (1998). *Predictive data mining-A practical approach.* Morgan Kaufmann Publishers.

Wiener, E., Pedersen, J.O., & Weigend, A.S. (1995). A neural network approach to topic spotting. *Fourth Annual Symposium on Document Analysis and Information Retrieval.*

Wu,G., & Chang,E. (2003). Class-Boundary Alignment for Imbalanced Dataset Learning. *Workshop on Learning from Imbalanced Data Sets(ICML).* Washington, DC.

Wu, J., Xiong, H., Wu, P., & Chen, J. (2007). Local decomposition for rare class analysis. *Proc. KDD*, (pp. 814-823).

Xue,G., Xing, D., Yang,Q., & Yu,Y. (2008). Deep Classification in Large-scale Text Hierarchies. *SIGIR.*

Yang, Y. (1999). An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval* , 69-90.

Yang, Y.,& Liu, X. (1999). A re-examination of text categorization methods. *SIGIR* , 42-49.

Yang,Y., & Chute,C.G. (1994). An example-based mapping method for text categorization and retrieval. *ACM Transaction on Information Systems*, (pp. 253–277).

Yang,Y.,& Pederson,J.O. (1997). A comparative study on feature selection in text categorization. *Proc. of the 14th international conference on machine learning*, (pp. 412-420).

Yang.Y. (2001). A study on thresholding strategies for text categorization. Proceedings of the 24th annual international ACM SIGIR conference on research and development, (pp. 137–145).

Zelikovitz, S.,& Hirsh,H. (2000). Improving short text classification using unlabeled background knowledge. *Proceedings of the Seventeenth International Conference on Machine Learning.*

Zhao, J.H., Li,X., & Dong,Z.Y. (2007). Online rare events detection. *Advances in Knowledge Discovery and Data Mining, Springer.*, pp. 1114-1121.

Zhou, Z.,H., & Liu, X.Y. (2006). On Multi-Class Cost-Sensitive Learning. *Proc. Nat'l Conf. Artificial Intelligence*, (pp. 567-572).

Zhuang, L., & Dai, H. (2006). Parameter Estimation of One-Class SVM on Imbalance Text Classification. *Lecture Notes in Artificial Intelligence*, pp. 538-549.

Zhuang, L., & Dai, H. (2006). Parameter Optimization of Kernel-Based One-Class Classifier on Imbalance Text Learning. *Lecture Notes in Artificial Intelligence*, pp. 434-443.