Fall 12-8-2016

Improving the Prediction Accuracy of Text Data and Attribute Data Mining with Data Preprocessing

PRIYANGA CHANDRASEKAR
Kennesaw State University

IMPROVING THE PREDICTION ACCURACY OF TEXT DATA AND ATTRIBUTE DATA MINING WITH DATA PREPROCESSSING

A Thesis Presented to

The Faculty of the Computer Science Department

by

Priyanga Chandrasekar

In Partial Fulfillment

of Requirements for the Degree

Masters in Computer Science

Kennesaw State University

Fall 2016

IMPROVING THE PREDICTION ACCURACY OF TEXT DATA AND ATTRIBUTE DATA MINING WITH DATA PREPROCESSSING

Approved:

Professor Dr. Chia-Tien Dan Lo, Committee Chair
Department of Computer Science
Kennesaw State University

Professor Dr. Kai Qian, Advisor
Department of Computer Science
Kennesaw State University

Professor Dr. Yong Shi,
Department of Computer Science
Kennesaw State University

In presenting this thesis as a partial fulfillment of the requirements for an advanced degree from Kennesaw State University, I agree that the university library shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish, this thesis may be granted by the professor under whose direction it was written, or, in his absence, by the dean of the appropriate school when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from or publication of, this thesis which involves potential financial gain will not be allowed without written permission.

_____

Priyanga Chandrasekar

Notice To Borrowers

Unpublished theses deposited in the Library of Kennesaw State University must be used only in accordance with the stipulations prescribed by the author in the preceding statement.

The author of this thesis is:

Priyanga Chandrasekar

1100 South Marietta Pkwy,
Marietta, GA 30060

The director of this thesis is:

Dr. Chia-Tien Dan Lo

1100 South Marietta Pkwy,
Marietta, GA 30060

Users of this thesis not regularly enrolled as students at Kennesaw State University are required to attest acceptance of the preceding stipulations by signing below. Libraries borrowing this thesis for the use of their patrons are required to see that each user records here the information requested.

IMPROVING THE PREDICTION ACCURACY OF TEXT DATA AND ATTRIBUTE DATA MINING WITH DATA PREPROCESSSING

An Abstract of

A Thesis Presented to

The Faculty of the Computer Science Department

by

Priyanga Chandrasekar

Bachelor of Technology, SASTRA University, 2010

In Partial Fulfillment

of Requirements for the Degree

Masters in the Computer Science

Kennesaw State University

Fall 2016

ABSTRACT

Data Mining is the extraction of valuable information from the patterns of data and turning it into useful knowledge. Data preprocessing is an important step in the data mining process. The quality of the data affects the result and accuracy of the data mining results. Hence, Data preprocessing becomes one of the critical steps in a data mining process.

In the research of text mining, document classification is a growing field. Even though we have many existing classifying approaches, Naïve Bayes Classifier is good at classification because of its simplicity and effectiveness. The aim of this paper is to identify the impact of preprocessing the dataset on the performance of a Naïve Bayes Classifier. Naïve Bayes Classifier is suggested as the best method to identify the spam emails. The Impact of preprocessing phase on the performance of the Naïve Bayes classifier is analyzed by comparing the output of both the preprocessed dataset result and non-preprocessed dataset result. The test results show that combining Naïve Bayes classification with the proper data preprocessing can improve the prediction accuracy.

In the research of Attributed data mining, a decision tree is an important classification technique. Decision trees have proved to be valuable tools for the classification, description, and generalization of data. J48 is a decision tree algorithm which is used to create classification model. J48 is an open source Java implementation of the C4.5 algorithm in the Weka data mining tool. In this paper, we present the method of improving accuracy for decision tree mining with data preprocessing. We applied the supervised filter discretization on J48 algorithm to construct a decision tree. We compared the results with the J48 without discretization. The results obtained from experiments show that accuracy of J48 after discretization is better than J48 before discretization.

IMPROVING THE PREDICTION ACCURACY OF TEXT DATA AND ATTRIBUTE DATA MINING WITH DATA PREPROCESSSING

A Thesis Presented to

The Faculty of the Computer Science Department

by

Priyanga Chandrasekar

In Partial Fulfillment

of Requirements for the Degree

Masters in Computer Science

Advisor: Dr. Kai Qian

Kennesaw State University

Fall 2016

DEDICATION

This thesis is dedicated to my family and my husband.

I thank everyone who supported me throughout my journey.

PREFACE

The thesis is submitted in partial fulfillment of the requirements for the master's degree at the Kennesaw State University, Kennesaw.

The research project has been conducted under the supervision of Professor Dr. Kai Qian, Department of Computer Science, during the years 2015-2016. Financing for the work has been provided in the form of scholarship from the Kennesaw State University through the Graduate Research Assistantship.

ACKNOWLEDGEMENTS

I would like to thank Dr. Kai Qian for his support, encouragement and motivation through this entire process.

I would also like to thank my thesis committee members, Dr. Chia-Tien Dan Lo and Dr. Yong Shi for their insightful comments and valuable suggestions.

This research paper is made possible through the help and support from everyone, including my professors, parents, my husband, family and friends.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Because of large amount of features in the dataset, properly identifying the documents into specific category poses various challenges. Being a popular way for communication, Email is more prone to misuse. In the electronic messaging systems, spam is used to send unsolicited bulk messages to many recipients. The amount of incoming spam increases every day. The spammer spread harmful message and even virus. The spammer creates spam in such a way that it looks like a normal message in order to avoid being detected. Sometimes the spam is nothing but a simple plain text with a malicious URL or some is clustered with attachments and/or unwanted images. Text based classifiers are used to find and also to filter spam emails.

Text classification is one of the main cores of our work. We used the supervised learning method called Naïve Bayes classifier. It can be programmed in the map reduce model. The separation of spam email from the ham email can be done more efficiently with the help of the data mining. With the knowledge gained from training phase, Bayesian classifier identifies the spam email from ham email. In the training phase, the emails are manually classified as spam or ham and the required features are set here.

Proposed by the Songtao [1], Map reduce model of the Naïve Bayes classifier proved to be effective when dealing with the huge data. Naïve Bayes Classifier is a Probabilistic classifier. In the machine learning, Naïve Bayes classifiers are highly scalable and it is also a popular method for text categorization with the word frequencies as the features. It assumes that features are independent. This content based classifier was proved to be efficient by Sahami [2].

Hadoop, an open source software framework is used in our paper. Hadoop is a scalable, distributed computing system which is capable of handling large amount of data. It consists of two main elements: MapReduce and Hadoop Distributed File System (HDFS). Hadoop breaks down the large dataset into multiple partitions and process them in parallel.

Data mining is the process of extracting useful information and knowledge from the incomplete, noisy and inconsistent raw data. Data mining extracts information from large dataset and converts it to an understandable form. Data mining is a part of knowledge discovery process. Classification is a form of data analysis that extracts model describing important data classes. Those models are called classifiers; predict categorical class labels. For example, a classification model can be built to categorize bank loan applications as either safe or risky [3].

Decision tree induction is the process of learning of decision trees from class labeled training tuples. Decision tree is an algorithm which is commonly used to predict model, and also to find out the valuable information through the huge amounts of data classification. A decision tree is a simple flowchart like tree structure, where the topmost node in a tree is the root node [4]. Each leaf node (or terminal node) holds a class label, each internal node (non-leaf node) denotes a test on an attribute, and each branch represents an outcome of the test. J48 is an extension of ID3. The additional features of J48 are accounting for missing values, decision trees pruning, continuous attribute value ranges, derivation of rules, etc.

The remaining of this paper is organized as follows. Section 2 gives a brief note about the related works. Section 3 presents and discusses our research methodology followed by the description of the Enron and microarray dataset we have used in our experiments as well as the experimental setup. In Section 4, the evaluation of results along with the performance analysis is presented. Finally, in

Section 5 our conclusions are presented followed by the references.

CHAPTER II

RELATED WORK

In this section we summarize the previous related research work on data preprocessing of both text data and attribute data.

Since 1993, Electronic mail is commonly known as email or e-mail. A way of exchanging messages from an individual to one or more recipients is called E-mail. Email operates across the Internet or other computer networks . In 2013, Total email sent and received per day was 182.9 billion. In that, the number of business emails sent and received per day was nearly 100.5 billion and the number of consumer/personal emails sent and received per day were 82.4 billion. [5]

Naive Bayes classifiers use Bayes' theorem to calculate a probability that an email is or is not spam. The Spam message usually consists of plain text. But in order to avoid being detected by the spam filter, the spammers make it more complicated with image and other attachments. There are different algorithms exist to find the different styles in the spam. To find the spam message with the images, NDD, SIFT and TR-FILTER is available. Ketari propose the major image spam filtering techniques [6]. The survey of the various kinds of algorithms is explained by Deshmukh [7].

Being one of the hottest internet issues, Spam email issue has been already addressed by many researchers. They have proposed a number of methods to deal with spam detection based on machine learning algorithms. Among them, Naïve Bayes classifier is suggested as a more effective method, which is a text-based classifier. Our study focuses mainly on the importance of preprocessing the dataset and also on how preprocessing helps to improve the accuracy.

For surveying the problem of improving decision tree classification algorithm for large attribute data sets, several algorithms have been developed for building DTs of large data sets. Kohavi & John 1995 [8], searched for parameter settings of C4.5 decision trees that would result in optimal performance on a particular data set. The optimization objective was "optimal performance" of the tree, i.e., the accuracy measured using 10-fold cross-validation. J48, Random Forest, Naive Bayes etc. algorithms [9] are used for disease diagnosis as they led to good accuracy. They were used to make predictions. The dynamic interface can also use the constructed models that mean the application worked properly in each considered case.

The classification algorithms [10] Naive Bayes, decision tree (J48), Sequential Minimal Optimization (SMO), Instance Based for K-Nearest neighbor (IBK) and Multi-Layer Perception are compared by using matrix and classification accuracy. Three different breast cancer databases have been used and classification accuracy is presented on the basis of 10-fold cross validation method. A combination at classification level is accomplished between these classifiers to get the best multi-classifier approach and accuracy for each data set. Diabetes and cardiac diseases [11] are predicted using Decision Tree and Incremental Learning at the early stage.

Liu X.H 1998 [12], proposed a new optimized algorithm of decision trees. On the basis of ID3, this algorithm considered attribute selection in two levels of the decision tree and the classification accuracy of the improved algorithm had been proved higher than ID3. Liu Yuxun & Xie Niuniu 2010 [13], solving the problem of a decision tree algorithm based on attribute importance is proposed. The improved algorithm uses attribute-importance to increase the information gain of attributes which has fewer attributions and compares ID3 with improved ID3 by an example. The experimental analysis of the data shows that the improved ID3

algorithm can get more reasonable and more effective rules. Gaurav & Hitesh 2013 [14], propose C4.5 algorithm which is improved by the use of L‟Hospital Rule, this simplifies the calculation process and improves the efficiency of decision making algorithms.

Though many researchers already studied the J48 classifier, we focused on improving the accuracy of the results. In our study, we applied the preprocessing-discretization on the J48 algorithm.

CHAPTER III

RESEARCH METHODOLOGY

3.1. Text Data Methodology

In our experiment, text dataset methodology has two phases: training and classification. The dataset is a known corpus. The count of occurrence of tokens was taken by map reduce model of Hadoop. With this count, knowledge about the dataset is learned. This knowledge is used in the classification phase to identify the spam probability in the new email set.

3.1.1 Dataset

Enron's dataset [15] which consists of 4500 spam emails and 1500 ham emails is used as training dataset. The dataset is manually labeled as ham or spam and it does not have encoding in it. Testing dataset which consists of 270 ham emails and 330 spam emails is used to test. In order to have better accuracy in the result, the test dataset was not included in the training dataset and it acts as "unknown" data. A sample message in the training dataset is shown in the figure 1.

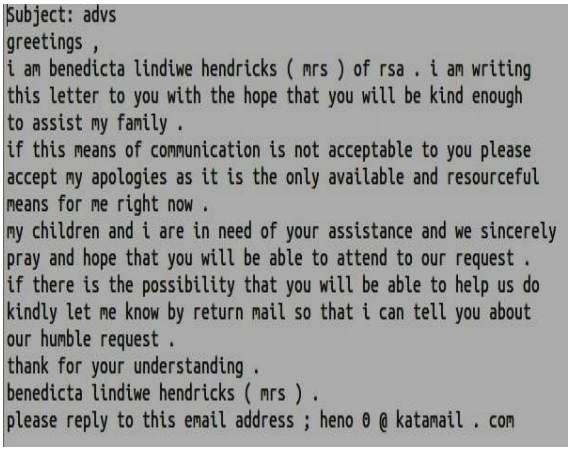

Figure 1: Sample Email content

3.1.2. Preprocessing

Real world data are generally incomplete, noisy and inconsistent. Data preprocessing is a first step of the Knowledge discovery in databases (KDD) process. Data preprocessing is a challenging and tedious task. There are number of different tools and methods available for data preprocessing. The tasks in the data preprocessing are: Data Cleaning, Data Integration, Data Transformation, Data Reduction, and Data Discretization.

Among those methods, we have used Data cleaning and Data reduction on the Naïve Bayes classifier. The following Preprocessing methods will make the dataset more precise. Hence the performance of the Naïve Bayes classifier will be more accurate and also the processing time will be reduced. Those data preprocessing methods are noisy removal, feature extraction and attribute reduction.

Noisy Removal: Some words contribute less in determining the email as spam or legitimate. Those words can be excluded in this step which will improve the efficiency of the classifier.

Feature Extraction: Feature extraction is one of the most important preprocessing steps. In this step, we found out all emails from dataset and replaced it with the term “EmailC”. In this way, the possible combination of attributes will be combined into the single subset. Similarly, all links from the dataset will be found and replaced with the term “URLC”. Hence the data will become more precise.

Stemmer: By using Stanford’s API [16] for English words lemmatization will reduce size of features and also processing time. For example, “earn”, “earned” and “earning” should be considered as single feature “earn”.

3.1.3. Training

The general processing of Naïve Bayes classifier can be described as follows: get a labeled sample and train it to build up the probabilities of each token in corpus, then the word probability obtained in the previous step would be used to compute the probability that an email with a particular set of words in it belongs to either category.

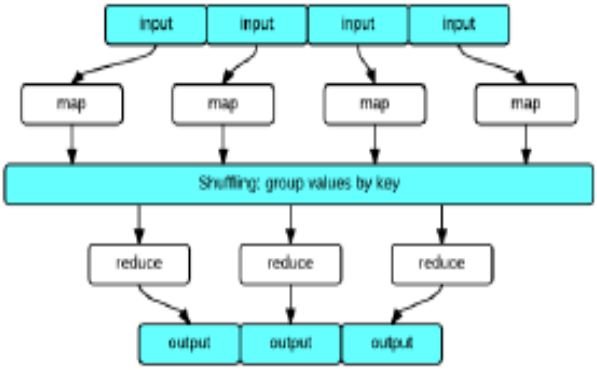

Figure 2: Map reduce Framework

In the Training Phase, the word count of the sample dataset is taken. The training dataset is already classified as ham and spam emails. In order to process huge dataset, Hadoop Map reduce framework is used in our work. The sample dataset is first being uploaded to HDFS, then split into independent chunks being processed by different map tasks, which would emit <key, value> pairs as output. The output here becomes the input of reduce task and the result would be stored in HDFS. The Hadoop map reduce processing flow is shown in the figure 2. Part of the training result is shown in figure 3, the first column represents the word in the dataset and the second column is the number of occurrences of that word.

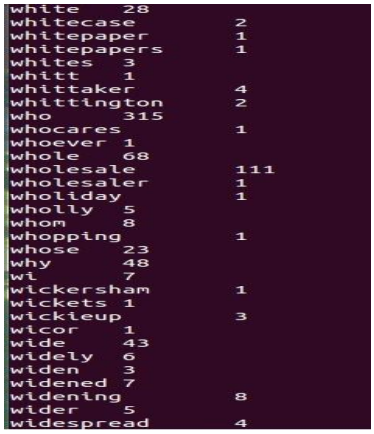

Figure 3: Word count from the dataset

3.1.4. Classification

The formula used to calculate the spamicity of a word is

$$P(S/W) = P(W/S)*P(S)/(P(W/S)*P(S)+P(W/H)*P(H))$$

where P(S/W) denotes the probability that a message is spam with the word W in it; P(S) is the overall probability that any given message is spam; P(W/S) is the probability that a particular word appears in spam messages; P(H) is the overall probability that any given message is ham; and P(W/H) is the probability that a particular word appears in ham messages.

Most Bayesian spam filtering algorithms are based on formulas that hold strictly only if the words present in the message are independent of each other, which is not always satisfied (for example, in natural languages like English the probability of finding an adjective is affected by the probability of having a noun), but it is a useful idealization, especially since the statistical correlations between the individual words are usually not known. On this basis, one can apply the Bayes' theorem to calculate the probability that a message is a spam with the bag of words in it.

$$P = P_1P_2 \ldots P_N / (P_1P_2 \ldots P_N + (1-P_1)(1-P_2)\ldots(1-P_N))$$

Where, P_1, P_2 ... P_N are the probabilities that a message is spam-knowing words.

3.2 Attribute Data Methodology

Our methodology is to learn about the dataset, apply J48 decision tree classification algorithms and get the accuracy of the algorithm. In preprocessing step, apply the supervised discretization filter on the dataset along with the J48 classification algorithm and find the accuracy. Finally comparing both accuracy and find out which one is better.

3.2.1 Leukemia Dataset

In our study we have used a real world leukemia microarray experiment performed by [Golub et al. 1999]. Leukemia is a cancer of bone marrow or blood cells. In general, leukemia's can be grouped into four categories. Myeloid and lymphoid leukemia's can be acute or chronicle whereas myeloid and lymphoid both denote cell types involved. Thus, four main types of leukemia's are: Acute Myeloid Leukemia (AML), Chronic Myeloid Leukemia (CML), Acute Lymphoblastic Leukemia (ALL) and Chronic Lymphoblastic Leukemia (CLL).

In the dataset provided by [Golub et al. 1999], each microarray experiment corresponds to a patient (example); each example consists 7129 genes expression values (features). Each patient has a specific disease (class label), corresponding to two kinds of leukemia (ALL and AML). There are 72 patients (47 ALL and 25 AML). The original study of [Golub et al. 1999] split patients into two disjoint sets: the training set contains 38 examples (27 ALL and 11 AML) and the test set contains 34 examples (20 ALL and 14 AML). Considering the shortage of examples it is a common technique in machine learning to use cross-validation or bootstrap [Kohavi 1995, Hastie et al. 2001] rather than isolating training and test sets.

In our study, training dataset participates in the test dataset. Hence our study uses the training set which contains 38 examples (27 ALL and 11 AML) and the test set which contains 38 examples (27 ALL and 11 AML).

3.2.2 WEKA

Weka is open-source software developed at the University of Waikato and the programming language is based on Java. Weka has 4 different applications, Explorer, Experimenter, KnowledgeFlow and Simple CLI. Knowledge Flow is a node and linked based interface and Simple CLI is the command line prompt version where each algorithm is run by hand. In our study, we used Explorer applications of the Weka.

WEKA is an innovatory tool in the history of the data mining and machine learning research communities. By putting efforts since 1994 this tool was developed by WEKA team. WEKA contains many inbuilt algorithms for data mining and machine learning. Weka implements algorithms for data preprocessing, classification, regression, clustering, association rules; it also includes a visualization tools.

3.2.3 J48 Classifier

Classification is the process of assigning an appropriate class label to an instance (record) in the dataset. Classification is generally used in supervised datasets where there is a class label for each instance. In our study we applied J48 classifier in the dataset. J48 Classifier uses the normalized version of Information Gain which is the Gain Ratio for building trees as the splitting criteria. It has both reduced error pruning and normal C4.5 pruning option. In our experiments we have used the algorithm J48 (with default parameters) from Weka [Witten and Frank 2005], a library of several machine learning algorithms. J48 is a Java implementation of the well-known C4.5 algorithm [Quinlan 1993]. J48 uses a modified version of the entropy measure from information theory.

3.2.4 Pre-processing

Data usually comes in mixed format: nominal, discrete, and/or continuous. Discrete and continuous data are ordinal data types having orders among values, while nominal values do not possess any order amongst them. Discrete data are spaced out with intervals in a continuous spectrum of values. We used discretization as data preprocessing method.

Discretization: Discretization process will easily interpret numerical attributes turning into nominal (categorical) ones. This process is done by dividing a continuous range into subgroups. Suppose there are 200 people in a group that want to apply for a bank loan and their ages are between 20 and 80. If the bank workers want to categorize them, they have to put them into some groups. For example one can categorize people between 20 and 40 as young, people between 40 and 65 as middle aged and 65 to 80 as old. So there will be three subgroups, which are; young, middle-aged and old. These subgroups can be increased depending on the choice of the field expert. This makes it easy to understand and easy to standardize.

Discretization of continuous attributes is both a requirement and a way of performance improvement for many machine learning algorithms. The main benefit of discretization is that some classifiers can only work on the nominal attributes, but not numeric attributes. Another advantage is that it will increase the classification accuracy of tree and rule based algorithms that depend on nominal data.

Discretization can be grouped into two categories, Unsupervised Discretization and Supervised Discretization. As the name implies Unsupervised Discretization is generally applied to datasets having no class information. The types of Unsupervised Discretization are: Equal Width Binning, Equal Frequency Binning mainly but more complex ones are based on clustering methods [17]. Supervised

Discretization techniques as the name suggests takes the class information into account before making subgroups. Supervised methods are mainly based on Fayyad-Irani [18] or Kononenko [19] algorithms.

Weka uses Fayyad-Irani method as default, so in our study we used Fayyad-Irani Discretization method. Weka has the Discretization algorithm under the preprocessing tab. As shown in Figure 4, it is embedded right under supervised and attribute options. Fayyad-Irani Discretization method is a supervised hierarchical split method, which will use the class information entropy of candidate partitions to select boundaries for discretization.

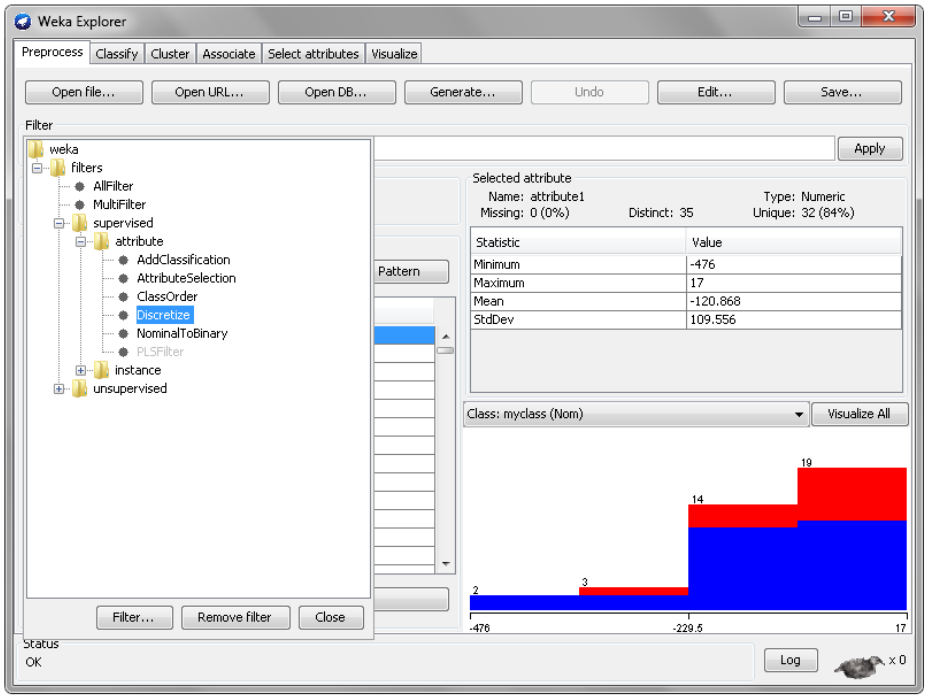

Figure 4: Selecting Discretization from Preprocess Tab

Class information entropy is a measure of purity and it measures the amount of information which would be needed to specify to which class an instance belongs. It considers one big interval containing all known values of a feature and then recursively partitions this interval into smaller subintervals until an optimal number of intervals are achieved.

One of the supervised discretization methods, introduced by Fayyad and Irani, is called entropy based discretization. The supervised discretization methods handle sorted feature values to determine the potential cut points such that the resulting cut point has the strong majority of one particular class. The cut point for discretization is selected by evaluating the favorite disparity measure (i.e., class entropies) of candidate partitions. In entropy based discretization, the cut-point is selected according to the entropy of the candidate cut-points. Entropy is defined as follows:

$$Entropy(D_1) = -\sum_{i=1}^{m} p_i \log_2 p_i$$

The Entropy Gain refers to how much entropy you gain by splitting a data set into two bins. Entropy Gain performs splits that maximize the improvement to the information we get from our data. Gain is defined as

$$Gain(E_{new}) = E_{initial} - E_{new}$$

CHAPTER IV

EVALUATION

4.1 Text Data Evaluation

While Evaluating the Naïve Bayes Classifier, We need to concentrate on four states for any data. Those states are true positive, true negative, false positive and false negative. A false positive means identifying legitimate email as spam. A false negative means identifying the spam as legitimate email. A false positive can have more impact than the false negative, since the users will miss the legitimate email content. The Environment used in our work is shown in the table 1.

Table 1: Environment Setup

	Name	Version
Operating System	Ubuntu	14.04
Java	Java SDK	7.0
IDE	Eclipse	For Linux
Big Data Analysis Framework	Hadoop	2.3.1

The Test dataset which consists of 270 ham emails and 330 spam emails was tested. In order to better accuracy in the result, the test dataset was not included in the training dataset and it act as “unknown” data. The evaluation of the result with the preprocessing of the dataset is shown in table 2.

Table 2: Evaluation Result with preprocessing

	Result	
	Result of spam test data	***Result of ham test data***
Total	330	270
Classified as spam	298	47
Precision	90.30%	82.59%
False positive	N/A	17.41%
False negative	9.69%	N/A

The evaluation of the result without preprocessing of the dataset is shown in table 3.

Table 3: Evaluation result without preprocessing

	Result	
	Result of spam test data	***Result of ham test data***
Total	330	270
Classified as spam	300	63
Precision	90.90%	76.67%
False positive	N/A	23.33%
False negative	9.09%	N/A

The comparison of both output shows improved precision and also false positives were greatly reduced for the preprocessed dataset. Thus the above test results shows that combining Naïve Bayes classification with the proper data pre-processing can improve the prediction accuracy and also proves that the preprocessing phase has a larger impact in the performance of the Naïve Bayes classifier especially with the reduced number of false positives.

4.2 Attribute Data Evaluation

While Evaluating the J48 classifier, we need to concentrate on false positive and false negative. A false positive means positive instances that are incorrectly assigned to the negative class. A false negative means negative instances that are incorrectly assigned to the positive class. A false positive can have more impact than the false negative. Initial experiment was to investigate the effect of discretization to the learning time and prediction accuracy of the J48 classifier. To figure that out, we need to run the algorithm on the dataset without discretization. Then we need to apply discretization and find out the results and compare the accuracy of the both.

A confusion matrix contains information about actual and predicted classifications done by a classification system. The Confusion matrix for training dataset without/with preprocessing is shown in Figure 5 and the confusion matrix for test dataset without/with preprocessing is shown in Figure 6.

```
=== Confusion Matrix ===

  a  b   <-- classified as
 23  4 |  a = ALL
  2  9 |  b = AML
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 24  3 |  a = ALL
  2  9 |  b = AML
```

Figure 5: Confusion matrix for training dataset without/with preprocessing

```
=== Confusion Matrix ===

  a  b   <-- classified as
 25  2 |  a = ALL
  6  5 |  b = AML
```

```
=== Confusion Matrix ===

  a  b   <-- classified as
 26  1 |  a = ALL
  3  8 |  b = AML
```

Figure 6: Confusion matrix for test dataset without/with preprocessing

Evaluation was carried out in the test dataset, which consists of 38 examples (27 ALL and 11 AML). The result shows that for the preprocessed dataset, accuracy of the decision tree was increased. The screenshots of the Weka result (Figure 5 and Figure 6) for the training and test dataset without/with discretization clearly shows that the accuracy of the J48 classifier improved when data was discretized.

Performance Analysis of J48 Classifier: The accuracies obtained by combining J48 Classification without discretization and with discretization were carried in both training and test dataset. The accuracies obtained were charted in Figure 7 for analysis.

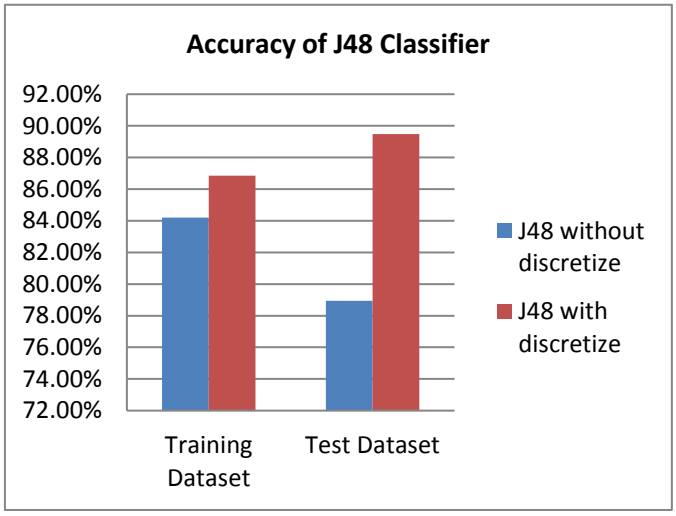

Figure 7 : Performance Analysis

CHAPTER V

CONCLUSION

In this paper, we added a pre-processing phase while training, which does noisy removal, extracts some typical features, and help improve the accuracy of email classification. With the training result, we achieved a moderate prediction when encountering a new incoming email. On the other hand, we did not preprocess the dataset and get the output. The comparison of both output shows improved precision and also false positives were greatly reduced by 25.39% for the preprocessed dataset. Thus the test results shows that combining Naïve Bayes classification with the proper data pre-processing can improve the prediction accuracy and also proves that the preprocessing phase has a larger impact in the performance of the Naïve Bayes classifier especially with the reduced number of false positives.

The first step of Data Mining, preprocessing process showed its benefits during the classification accuracy performance tests. In this paper, entropy-based discretization method is used for improving the classification accuracy for datasets including continuous valued features. In the first phase, the continuous valued features of the given dataset are discretized. Second phase, we tested the performance of this approach with the J48 classifier and compared with performance of J48 classifier without discretization.

Discretization of the numerical attributes increased the performance of J48 by approximately 2.63% for training dataset and 10.53% for test dataset. The result proves that the optimal level of discretization improves both the model construction time and prediction accuracy of the J48 classifier. Other benefit of discretization came after the visualization of J48, making the tree easy to

interpret, because of the cutting-points it assigned after the discretization of numerical attributes. Thus the test results shows that combining J48 classifier with the proper data pre-processing can improve the prediction accuracy and also proves that the preprocessing phase has a larger impact in the performance of the J48 classifier.

APPENDIX A

SOME ANCILLARY STUFF

If you would like to learn more about this research project, you can examine the following references in the next page that are referred in this work.

REFERENCES

[1] Songtao Zheng "Naïve Bayes Classifier – A MapReduce Approach" M.S. thesis, CS, NDSU, Fargo, N.D, 2014.

[2] M. Sahami, S. Dumais, D. Heckerman., and E. Horvitz,, "A Bayesian approach to filtering junk email," Proc. AAAI Workshop on Learning for Text Categorization, 1998, AAAI Technical Report WS-98-05.

[3] Mehmed Kantardzic, "Data Mining: Concepts, Models, Methods, and Algorithms", ISBN: 0471228524, John Wiley & Sons, 2003.

[4] Sushmita Mitra, & Tinku Acharya, "Data Mining Multimedia, Soft Computing, and Bioinformatics", John Wiley & Sons, Inc, 2003.

[5] http://sourcedigit.com/4233-much-email-use-daily-182-9-billion-emails-sentreceived-per-day-worldwide/

[6] L.M. Ketari, L..M. Chandra, and M. A. Khanum. "A Study of Image Spam Filtering Techniques," 4th IEEE Internat. Conf. Computational Intelligence and Communication Networks, 2012.

[7] S.S. Deshmukh., P.R. Chandre, "Survey on: Naive Bayesian and AOCR Based Image and Text Spam Mail Filtering System", International Journal of Emerging Technoogy and Advanced Enginerring.

[8] Tea Tusar, "Optimizing Accuracy and Size of Decision Trees", Department of Intelligent Systems, Jozef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia, 2007

[9] Robu, R.; Hora, C., "Medical data mining with extended WEKA," Intelligent Engineering Systems (INES), 2012 IEEE 16th International Conference on , vol., no., pp.347,350, 13-15 June 2012

[10] Salama, G.I.; Abdelhalim, M.B.; Zeid, M.A., "Experimental comparison of classifiers for breast cancer diagnosis," Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on , vol., no., pp.180,185, 27-29 Nov.,2012.

[11] UM, Ashwinkumar, and Anandakumar KR. "Predicting Early Detection of Cardiac and Diabetes Symptoms using Data Mining Techniques.",IEEE,pp:161-165,2011

[12] Weiguo Yi, Jing Duan, &Mingyu Lu, "Optimization of Decision Tree Based on Variable Precision Rough Set", International Conference on Artificial Intelligence and Computational Intelligence, 2011.

[13] Liu Yuxun, &XieNiuniu, “Improved ID3 Algorithm”, IEEE, 2010.

[14] Gaurav L. Agrawal, & Prof. Hitesh Gupta, “Optimization of C4.5 Decision Tree Algorithm for Data Mining Application”, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 3, March 2013.

[15] http://www.aueb.gr/users/ion/data/enron-spam/

[16] http://nlp.stanford.edu/software/corenlp.shtml/

[17] Joa˜o Gama and Carlos Pinto. Discretization from data streams: applications to histograms and data mining. In Proceedings of the 2006 ACM symposium on Applied computing, SAC ’06, pages 662–667, New York, NY, USA, 2006. ACM.

[18] Usama M. Fayyad and Keki B. Irani. Multi-interval discretization of continuousvalued attributes for classification learning. In Thirteenth International Joint Conference on Articial Intelligence, volume 2, pages 1022–1027. Morgan Kauf- mann Publishers, 1993.

[19] Igor Kononenko. On biases in estimating multi-valued attributes. In 14th Inter- national Joint Conference on Articial Intelligence, pages 1034–1040, 1995.

VITA

Priyanga Chandrasekar is currently a graduate student in Computer Science at Kennesaw State University. She received the undergraduate degree in Bachelor of Technology - Electronics and Communication Engineering from SASTRA University, Thanjavur and worked with Nokia Siemens Networks for 2 years as a Network Engineer.

Current Research Area: Data mining

Programming Interests: Data Analytics, Java and Android App Development.