

E-mail Classification System for Bank Internal mail System

K.T Atapattu

139155D



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of the Master Degree of Science in Information Technology.

April 2016

Declaration


I hereby certify that this project and the all the artefacts associated with it is my own work and it has not been submitted before nor is currently being submitted for any other degree programme.

Full name of the student: Kasun Tharanga Atapattu

Student Number : 139155D

Signature of the student:

Date:

 **Name of the supervisor:** Mr. Saminda Premaratne
University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Signature of the supervisor:

Date:

Dedication

This Dissertation is dedicated to my loving parents for being part of me and encouraging me always being by my side.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Acknowledgement

Apart from the efforts of me, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this Research.

Special mention goes to my supervisor, Mr. Saminda Premarathne not only for his tremendous academic support, but also for giving me opportunity to successfully complete the research.

My next thank goes to Prof. Asoka S Karunanda who taught us Research Methodology and Literature Review and Thesis Writing subjects which were more benefited us throughout the research work

It is my great pleasure to thank all the other Senior lecturers, Lecturers, Instructors, and staff members who helped us in many ways to make this Research. The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the Research. I am grateful for their constant support and help.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Then I would like to thank my all the batch mate of the M.Sc Information Technology batch 7 in faculty of Information Technology for their various help and support. All so other friend of the faculty as well as friends for outside who gave me supporting and encouraging us with their best wishes.

Abstract

My project is an implementation of 'Naive Based Algorithm' to classify E-mails. The main idea of the project is to implement algorithm in a effective manner to classify bank E-mails in bank internal authenticated mail system. E- mail has the potential to improve efficiency and reduce costs involved in communication. Even after the advent of newer technologies such as instant messaging and VoIP, email remains the number one application for business communication.

With the increasing of information on the internet based communication, our bank internal authentication mail system needs an efficient tool to classify the E-mails into categories. In this way, we can easily classify E-mail from large amount of E-mails available. Automated text categorization is a process that assigning pre-defined category labels to E-mail based on the contents.

Text categorization has many applications. For example, we can classify web pages into different categories to speed up the internet search, which is very useful for some search engines like Yahoo, Google etc. Also E-mail service providers are using those classification techniques for spam filtering and E-mail classification as well.

I have trained the develop algorithm from more than thousand pre - categorized E-mails. I have tested the text categorization algorithm developed based on naive based classifier with several different size data sets. Accuracy is evaluated as well. Experiment results shows my conclusion is efficient.

Contents

DECLARATION	I
DEDICATION.....	II
ACKNOWLEDGEMENT.....	III
ABSTRACT	IV
CHAPTER 1	1
INTRODUCTION.....	1
1.1 Background and Motivation.....	1
1.2 Aims and Objectives	2
1.2.1 Aims of the research.....	2
1.2.2 Objectives of the research.....	3
1.5 Structure of the Thesis.....	3
CHAPTER 2	4
LITERATURE REVIEW	4
2.1 Summary.....	9
CHAPTER 3	10
BANK CUSTOMER E-MAIL CLASSIFICATION.	10
3.1 Introduction	10
3.2 Factor Identification for Customer E-mail Classification.	10
3.3 Data Cleaning and Pre-processing	10
3.4 Extract the set of key words for each attribute in each of the factors.....	10
3.5 Use text classification technique to analyze the data.....	11
3.6 Machine Learning	11
3.6.1 Bayesian Filter	11



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

3.7 WordNet as ontology.....	12
3.8 Summary	12
CHAPTER 4.....	13
TEXT CLASSIFICATION TECHNIQUE FOR E-MAIL CLASSIFICATION.....	13
4.1 Introduction	13
4.2 Methodology Used for System Design and Development	13
4.3 Java	13
4.4 Oracle Database	13
4.5 Wordnet.....	14
4.5.1 Database content	14
4.6 Methodology.....	14
4.6.1 Category identification for the classification.	15
4.6.2 Retrieve the related data e-mail data set for the classification.....	16
4.6.3 Data Cleaning and pre-processing.....	16
4.6.3.1 Stop words.....	16
4.6.3.2 Pre - Processing.....	17
4.6.4 Develop classification using Bayes' Theorem.....	17
4.6.4.1 Bayes' Theorem.....	17
4.6.4.2 Train E-mail classification algorithm using Bayes' Theorem.....	18
4.6.4.3 Get probability of new e-mail from train algorithm.	20
4.7 Summary	21
CHAPTER 5.....	22
SYSTEMS DESIGN	22
5.1 Introduction	22
5.2 Architectural Design of the system.....	22
5.3 Pre - processing layer	23
5.3.1 Architectural Design of the pre - processing layer	24
5.4 Classification layer	25
5.4.1 Architectural Design of the classification layer	26

5.5 Summary	26
CHAPTER 6	27
IMPLEMENTATION	27
6.1 Introduction	27
6.2. Implementation of Pre - Processing Layer	27
6.2.1 Component for read the e-mails retrieved from bank authentication system to Microsoft excel document.	27
6.2.2 Component for remove stop words.	28
6.2.3 component for data pre - processing	29
6.2.4 Component for enter new e-mail to get the category and change the category if required.	30
6.3 Implementation of Classification Layer	32
6.3.1 Use WordNet for avoid repeating words with similar meaning.	32
6.3.2 Train the algorithm with pre - processed data and Calculate positive probability using Bayes' Theorem.	33
6.3.3 Classify new e-mail using trained algorithm and self train the algorithm from provided e-mail.....	36
6.4 Summary	37
CHAPTER 7	38
EVALUATION	38
7.1 Introduction	38
7.2 Evaluation	38
7.2.1 Component testing	38
7.2.1.1 Evaluate 'Read E-mails' Component	38
7.2.1.2 Evaluate ' Remove stop words' Component	39
7.2.1.3 Evaluate ' Data pre - processing' Component.....	39
7.2.1.4 Evaluate ' WordNet' Component	39
7.2.1.5 Evaluate ' Calculate Positive Probability' Component	39
7.2.1.6 Evaluate ' Classify new e-mail using trained algorithm' Component	40
7.2.1.7 Evaluate Self train the classification algorithm Component.....	40
7.2.2 Integration Testing	40
7.2.2.1 Implementation of Integration testing	40
7.3 Summary	41



CHAPTER 8	42
CONCLUSION	42
8.1 Introduction	42
8.2 Analysis of Performance and Accuracy.	42
8.2.1 Algorithm training performance.....	42
8.3 Algorithm drawbacks	43
8.4 Limitations	43
8.5 Future Work	43
8.6 Summary	44
REFERENCES	45



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Introduction

1.1 Background and Motivation

With the introduction of Electronic Data Interchange (EDI), most of the industries including public and private sector are using electronically communicated data to drive their company and industry. Where the electronic methods are used by millions of peoples around the world to communicate with each other and data is now available in most of the compatible formats, also the data is always being updated and new challenges occur continuously[1].

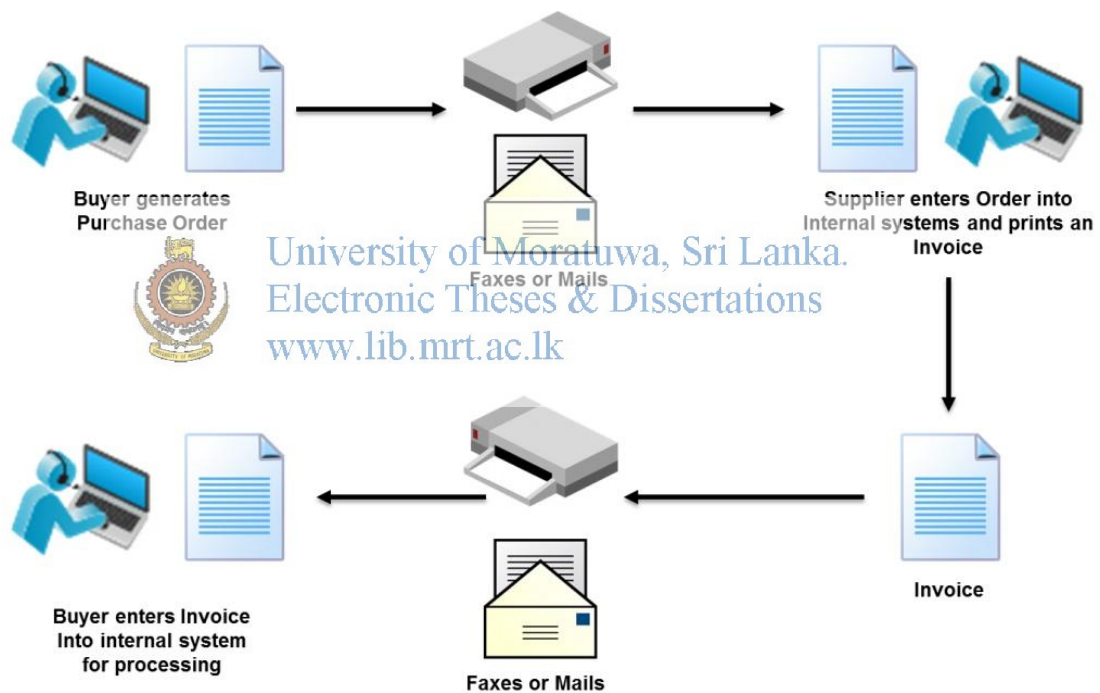


Figure 1

The continuously updated data can be used in many ways. For example the e-mail send to the bank by the customer can used to improve banking internal processes, new product development and trends. Most of the time customers are communicating with their bank via telephone call or E – mail. E – Mail can be categories in two areas in banking industry. They are general inquires receiving from bank corporate web site and authenticated mails receiving from bank customers via bank’s electronic delivery channel. Any person who visits the bank corporate web site can send their inquiry, complain or suggestion. [2] But authenticated mails are sending for a specific purpose

and targeting specific product in a bank. So these authenticated mails can be easily use for classification which can help to improve banking internal operations effectively.

Text categorization has many applications. For example, we can classify web pages into different categories to speed up the internet search, which is very useful for some search engines like Yahoo. In the bank existing authenticated mail system do not provide provision for the categorized the E-mails received from their online banking users. Text categorization can be a solution for that problem which is a incredible improvement of Internet and digital collection has caused a lot of research areas. Text categorization is a process that group text documents into one or more predefined categories based on their contents. It has wide applications, such as email filtering, category classification for search engines and digital libraries. Automatic Text Classification involves assigning a text document to a set of pre-defined categories, using a self learning technique.[3]

Basically there are two stages involved in text categorization. Training stage and testing stage. In training stage, documents are pre-processed and are trained by a learning algorithm to generate the classifier. In testing stage, a validation of classifier is performed.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

1.2 Aims and Objectives

1.2.1 Aims of the research

The aim of the project is to use the large amount of data available in the authenticated mail system, organizes provided data in to the covenant manner , filter and presents it in a way that is useable and relevant to the bank internal users.

The study would focus at efficient methods to extract meaningful words from bank authenticated mail system, analysis of data using different algorithm, identify the word occurrence probability using provided sample, train the provided algorithm by large amount of e-mails, and finally develop the system to classify new e-mail to the provided category in an effective manner.

1.2.2 Objectives of the research

- Effectively extract the data from E-mails
- Identify the factors that e-mails received to the bank authenticated mail system.
- Develop the classification algorithm using text classification method.
- Train the algorithm from large data sample to increase the accuracy.
- Develop system to classify the new received e-mail using training data.
- Train the algorithm from each new e-mail received while classification according the training data set.
- Allow user to change the e-mail type and re-train the algorithm from new user input.

1.5 Structure of the Thesis

This Chapter 1 provides an introduction to the email classification system and how it is important for bank consumers to categorize different types of emails they received from bank customers. Their aim and the objectives by doing this research is described in detail. Chapter 2 presents similar work done by others while Chapter 3 explains the technology adapted. Chapter 4 describes the approach of the research and Chapter 5 is about analysis and design. Chapter 6 is on implementation of the System. Finally, Chapter 7 presents the evaluation of the system and Chapter 8 concludes the results and suggests the further work to continue with this research, and list of references provides as the last section.

Chapter 2

Literature Review

Classification problem has been widely studied in areas of data classification, machine learning, databases and information research which contributed to number of industry applications, such as target marketing, medical diagnostics, filtering of newsgroup application and organizing documents. [5]. According to the research paper written by Charu C. Aggarwal The training data is used in order to construct a classification model, which relates the features in the underlying record to one of the class labels. For a given test instance for which the class is unknown, the training model is used to predict a class label for this instance.[5].He also mention different application types which are using text calcification algorithms. One of them is news filtering and Organization. Most of the news services today are electronic in nature in which a large volume of news articles are created every single day by the organizations. In such cases, it is difficult to organize the news articles manually. Therefore, automated methods can be very useful for news categorization in a variety of web portals. This application is also referred to as text filtering. Document Organization and Retrieval application is generally useful for many applications beyond news filtering and organization. A variety of supervised methods may be used for document organization in many domains. These include large digital libraries of documents, web collections, scientific literature, or even social feeds. Hierarchically organized document collections can be particularly useful for browsing and retrieval. As he mentioned in his research paper Opinion Classification and Spam Filtering are another important application in text classification. [5]. Furthermore he is pointing out about E-mail classification techniques as well. Even though he have used effective classification methods he did not adapt and 'Stopword' removal system to effectively classify the data.

The research conducted by Raj Kumar and Dr. Rajesh Verma pointing out the several different kind of classification algorithms currently in use such as C4.5, k-nearest neighbour classifier, Naive Bayes, SVM, Apriori, and AdaBoost.[6] In their research they are providing vital information about the C4.5 algorithm. C4.5 algorithm is Systems that construct classifiers are one of the commonly used tools in data classification. Such systems take as input a collection of cases, each belonging to one

of a small number of classes and described by its values for a fixed set of attributes, and output a classifier that can accurately predict the class to which a new case belongs. Like CLS and ID3, C4.5 generates classifiers expressed as decision trees, but it can also construct classifiers in more comprehensible rule set form.[6] He also mentioning about the limitations in C4.5 algorithm. First one is 'Empty branches'. Constructing tree with meaningful value is one of the crucial steps for rule generation by C4.5 algorithm. In our experiment, we have found many nodes with zero values or close to zero values. These values neither contribute to generate rules nor help to construct any class for classification task. Rather it makes the tree bigger and more complex. Second one is 'Insignificant branches'. Numbers of selected discrete attributes create equal number of potential branches to build a decision tree. But all of them are not significant for classification task. These insignificant branches not only reduce the usability of decision. Last one is 'Over fitting'.Over fitting happens when algorithm model picks up data with uncommon characteristics. This cause many fragmentations is the process distribution. Statistically insignificant nodes with very few samples are known as fragmentations . Generally C4.5 algorithm constructs trees and grows it branches 'just deep enough to perfectly classify the training examples'. This strategy performs well with noise free data. But most of the time this approach over fits the training examples with noisy data. Currently there are two approaches are widely using to by pass this over-fitting in decision tree learning .Those are if tree grows very large, stop it before it reaches maximal point of perfect classification of the training data and allow the tree to over-fit the training data then post-prune tree.[6].

Journal published by S.Neelamegam and Dr.E.Ramaraj point out vital information about another common algorithms like K-Nearest Neighbour, Support Vector Machines, Naive Bayesian Classification and Neural Networks.[7] According to them K-Nearest neighbour classifiers are based on learning by analogy. The training samples are described by n dimensional numeric attributes. Each sample represents a point in a n-dimensional space. In this way, all of the training samples are stored in a n-dimensional pattern space. When given an unknown sample, a k-nearest neighbour classifier searches the pattern space for the k training samples that are closest to the unknown sample. "Closeness" is defined in terms of Euclidean distance, where the Euclidean distance between two points,[7] and Support Vector Machine (SVM) very

effective method for regression, classification and general pattern recognition. It is considered a good classifier because of its high generalization performance without the need to add a priori knowledge, even when the dimension of the input space is very high. The aim of SVM is to find the best classification function to distinguish between members of the two classes in the training data. The metric for the concept of the "best" classification function can be realized geometrically. For a linearly separable dataset, a linear classification function corresponds to a separating hyper plane $f(x)$ that passes through the middle of the two classes, separating the two. [7]. They are pointing out the factors about Bayesian networks as well. Acyclic graph and a probability distribution for each node in that graph given its immediate predecessors . A Bayes Network (BN) Classifier is based on a Bayesian network which represents a joint probability distribution over a set of categorical attributes. It consists of two parts, the directed acyclic graph G consisting of nodes and arcs and the conditional probability tables. The nodes represent attributes whereas the arcs indicate direct dependencies. The density of the arcs in a BN is one measure of its complexity. Sparse BNs can represent simple probabilistic models (e.g., naïve Bayes models and hidden Markov models), whereas dense BNs can capture highly complex models. Thus, BNs provide a flexible method for probabilistic modelling.[7]. At last they mentioned about neural networks. An Artificial Neural Network (ANN), often just called a "neural network" (NN), is a mathematical model or computational model based on biological neural networks, in other words, is an emulation of biological neural system. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. [7]

In addition to above mention researches Vaibhav C.Gandhi, Jignesh A.Prajapati have done a Comparison between Text Classification Algorithms. The text categorization can be roughly classified as two classes, one is statistical based, e.g. Nave Bayes, the maximum Shannon entropy model, KNN, Support Vector Machine. The second is knowledge based classification method, e.g. Productive rules, neural network etc.[8]. To compare these methods in above algorithms in most cases, support vector machine (SVM) and K nearest neighbor (KNN) have better effect, neural network is after them, Naive Bayes is the last.[8]. But he did not mention the way to improve Naive Bayes Algorithm. This may be not the main objective of the researcher at that time.

Furthermore the applied research conducted by Patrick Ozer mentioning important information about the WEKA. WEKA is a collection of machine learning algorithms for Data Classification tasks. It contains tools for data classification, regression, clustering, association rules, and visualization. WEKA has four different modes to work in. They are Simple CLI provides a simple command-line interface that allows direct execution of WEKA commands, Explorer provides an environment for exploring data with WEKA, Experimenter provides an environment for performing experiments and conduction of statistical tests between learning schemes and Knowledge Finally Flow provides a “data-flow” inspired interface to WEKA. The user can select WEKA components from a tool bar, place them on a layout canvas and connect them together in order to form a “knowledge flow” for processing and analyzing data.[9]. But limitation is there was no pre-processing of the data.

The research work done by Trevor Mansuy and Robert J. Hilderman Evaluating WordNet Features in Text Classification Models, Incorporating semantic features from the WordNet lexical database is among one of the many approaches that have been tried to improve the predictive performance of text classification models.. [10]. Experimental results show that none of the WordNet relationships were effective at increasing the accuracy of the Naive Bayes classifier. Synonyms, hypernyms, and holonyms were effective at increasing the accuracy of the Coordinate Matching classifier, and hypernyms were effective at increasing the accuracy of the SVM classifier.[10]. They implemented a text classifier that can incorporate the various WordNet features into a category model. The instinct behind this is that keywords in the training set alone may not be extensive enough to enable generation of a universal model for a category. Another research done by Zakaria Elberrichi, Abdelattif Rahmoun, and Mohamed Amine Bentaalah is related to WordNet classification database. Text Categorization is the classification of documents with respect to a set of one or more pre-existing categories. Text Classification is a hard and very useful operation frequently applied to assign subject categories to documents, to route and filter texts, or as a part of natural language processing systems. [11]. In that paper, they have proposed a new approach for text categorization based on incorporating background Knowledge (WordNet) into text representation with using the X2 multivariate, which consists of extracting the K better features characterizing best the category compared to the others. But the main difficulty is that a word usually has multiple synonyms with somewhat different meanings and it is not easy to

automatically find the correct synonyms to use. [11]. However above two proved that WordNet is a vital database for find synonyms.

Handling Stop - words is challenge for text mining. The research done by Benjamin Klatt, Klaus Krogmann and Volker Kuttru suggesting an approach to develop reusable stop word lists to improve Natural Language Program Analysis. [12] They proposing to distinguish different scopes a stop word list applies to (i.e. programming language, technology, and domain) and recommend types of sources for terms to be included. Furthermore they are proposing an application of the concept as guidelines for developing stop word lists. [12]A another research conducted by Hassan Saif, Miriam Fernandez, Yulan He and Harith Alani for on Stopwords, Filtering and Data Sparsity for Sentiment Analysis of Twitter that Sentiment classification above Twitter is frequently affected by the noisy situation and they have identified that reducing this noisy nature will assist organisations and individuals to monitor the customer opinion to their brands and business.[13] Hence they need to apply 'stopwords removal' method to reduce the noise of textual data. They proved that according to their observations the use of pre-compiled 'stopword' list has a negative impact on the classification performance. Even though it is using widely in Twitter sentiment analysis. [13] However they did not compare this negative impact with other classification methods.

Limitations of earlier studies are described in Table 1

Limitation	Study
Not adapt and 'Stopword' removal system to effectively classify the data	[1]
Limitation of C 4.5 Algorithms for classification	[2]
Does not indicate the way to improve the Naive Bayes Algorithm	[4]
No preprocessing of data	[5]
Training set is not enough to make a universal model of classification	[6]

Table 1

2.1 Summery

The above mentioned researches are done using effective classification algorithms. They have limitations mentioned in the table 1. The problem identified was not indicate the way to improve the Naive Bayes Algorithm by using 'Stopword removal' methods or using 'Wordnet' database filter synonyms.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Bank Customer E-mail Classification.

3.1 Introduction

This chapter describes classification and modelling techniques adapted of the system. At the beginning this chapter will described about the algorithm developed.

3.2 Factor Identification for Customer E-mail Classification.

The approach that is considered for the model that is being used in this research is based on the customer profiling.

Bank customer authentication mail system is a mail system where bank customers communicating with the bank for spe cific purpose and those mails can catogirized into major categories. Bank customer e-mails can be categorized into following main categories

- Complain
 - Request
 - Other
- 
- University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

3.3 Data Cleaning and Pre-processing

Before data is submitted to the profiling data must be pre-processed. That means only essential data have extract from each e-mail before submitted to the model. This will increase the efficiency of algorithm and system can effectively process the data. Data pre - processing includes following two main sub processes

- Remove stop words : System will remove / avoid stop words which are not giving any weight age to classification.
- Filter meaningful words : System will select only meaningful words which are affecting the classification.

3.4 Extract the set of key words for each attribute in each of the factors.

The communication approach of the customer will be depending on the each factors mentioned in section 3.2. Each and every factor can be identified by specific key words. Based on the sample taken to the research we can identify key words for each factors. Sample mails can be identified for algorithm training. Those mails can be

manually categorized. Those categorized mails contained words which are unique for the given category. Each key word will be categorized for into each pre define category and train the developed algorithm from more e-mails which are related to given categories.

3.5 Use text classification technique to analyze the data.

After identifying essential key words, text classification technique can be introduced to identify and categorize the e-mails according to identified factors. After data cleaning and identifying the classification technique which is more suitable for the e-mail classification system, system can be developed for get best suitable solution for the identified problem and after provided training with large amount of data that developed system should be able to effectively categorized the given e-mail for the best effective category. Existing text classification algorithm with provided features which are appropriate for the e-mail classification can be provide effective result when it's come to the e-mail classification.

3.6 Machine Learning

Machine learning is make systems capable of learn from data. Learning in this context is not learning by heart but recognizing complex patterns and make intelligent decisions based on data. People have developed algorithms that can discover knowledge from specific data and experience, based on statistical and computational principles. Machine learning can be used to train the system to learn to distinguish between spam and ham messages. After learning phase, it can be used to classify SMSs. Machine Learning consists of probability theory, logic, combinatorial optimization, search, statistics, reinforcement learning and control theory. Vision to language processing, forecasting, pattern recognition, games, data mining, expert systems and robotics are some of the areas that use machine learning.

I am using feature selection and classification techniques under machine learning to train the filter as well as to improve the filtering capabilities.

3.6.1 Bayesian Filter

Bayesian spam filtering is a statistical technique that is mainly use in e-mail Cloassification. It uses naive Bayes classifier with content of words model which is a common technique in text classification features in order to identify related category. Naive Bayes classifiers compares tokens in the message against the pre-defined

categories and E-mails and then use Bayesian inference to calculate a probability of a message being a E-mail related to that category or not.

3.7 WordNet as ontology

Computer science and information science, ontology is a formal representation of knowledge as a set of concepts within a domain, and the relationships between those concepts. It is used to reason about the entities within that domain, and may be used to describe the domain. The hypernym/hyponym relationships among the noun synsets can be interpreted as specialization relations between conceptual categories. In other words, WordNet can be interpreted and used as a lexical ontology in the computer science sense.

3.8 Summary

This chapter briefly described about main idea about the project and how this can be linked to the banking sector and bank user behaviour. Next chapter will more elaborate about the methodology and the steps using for classification.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Chapter 4

Text Classification technique for E-mail classification.

4.1 Introduction

This chapter describes how to adopt classification and modelling techniques for bank customer behaviour profiling.

4.2 Methodology Used for System Design and Development

Object Oriented Analysis & Design Methodology (OOADM) is used for analyzing and designing the system, because it is easier to model real world objects using OOADM. For model the system I have used UML (Unified Modelling Language).

4.3 Java

I am using java as our base programming language to develop my system. Java is a powerful object-oriented programming language. It was designed for flexibility, allowing developers to write code that would run on any machine, regardless of architecture or platform. According to the Java home page, more than 1 billion computers and 3 billion mobile phones worldwide run Java. I have used NetBeans IDE for the implementations.

4.4 Oracle Database

Oracle database (Oracle DB) is a relational database management system (RDBMS) from the Oracle Corporation. Originally developed in 1977 by Lawrence Ellison and other developers, Oracle DB is one of the most trusted and widely-used relational database engines. The system is built around a relational database framework in which data objects may be directly accessed by users (or an application front end) through structured query language (SQL). Oracle is a fully scalable relational database architecture and is often used by global enterprises, which manage and process data across wide and local area networks. The Oracle database has its own network component to allow communications across networks.

Oracle Database provides support for developing, storing, and deploying Java applications. NetBeans IDE includes built-in support for Oracle Database. You can easily establish a connection from inside the IDE and begin working with the database

4.5 Wordnet

Standard alphabetical procedures for organizing lexical information put together words that are spelled alike and scatter words with similar or related meanings randomly through the list. Unfortunately, there is no obvious alternative, no other simple way for lexicographers to keep track of what has been done or for readers to find the word they are looking for. But a frequent objection to this solution is that finding things on an alphabetical list can be tedious and time-consuming. Many people who would like to refer to a dictionary decide not to bother with it because finding the information would interrupt their work and break their train of thought.

It is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short, general definitions, and records the various semantic relations between these synonym sets. WordNet database produce a combination of dictionary and thesaurus that is more intuitively usable, and to support automatic text analysis and artificial intelligence applications. The database can be downloaded and used freely. It can also be browsed online.

4.5.1 Database content

The database contains 155,287 words organized in 117,659 synsets for a total of 206,941 word-sense pairs. WordNet distinguishes between nouns, verbs, adjectives and adverbs because they follow different grammatical rules. Every synset contains a group of synonymous words or collocations. Different senses of a word are in different synsets. Most synsets are connected to other synsets via a number of semantic relations. [14] These relations vary based on the type of word, and include:

- Nouns
- Verbs
- Adjectives
- Adverbs

4.6 Methodology

Based on the above problem identification and the literature review done, supervised data classification method is proposed for the task of e-mail classification. The methodology mainly includes the basic steps that is specified below,

- Category identification for the classification.
- Retrieve the related data e-mail data set for the classification.
- Data Cleaning and Pre-processing.
- Develop classification using bayes' theorem.
- Identify similar words using WordNet database.
- Train the developed algorithm using existing database.
- Find the related category of the given e-mail from trained algorithm and train the algorithm by newly added e-mails
- Allow user to change the mail category required and train the algorithm.

4.6.1 Category identification for the classification.

The categories that affect for the categorization should be defined. There are various methods and models that are used in various applied areas for the purpose of classification.

Classification categories identification could be done through the understanding the banking process and the purpose of the online banking authentication mail system.

After studying the bank authentication mail processes I could identify three main areas where we can categorized those e-mails. They are described as follows

- **Request**

Normally bank customers user bank authentication mail system send their specific mail requests to their relationship managers such as open an accounts, close an accounts, change customer communication mail address, request details about bank specific products etc.

- **Complain**

As a service oriented industry banks are highly depend on customer service. Therefore bank relationship managers frequently getting customer complains about their service renders to customers and various issues in the new products specially electronic delivery channels. These mails need to be specially treated for the immediate action.

- **Other**

Other than above mentioned two categories bank customers may send mails to the authentication mail system such as their new suggestions, inquire about bank interest rates etc. They can be categorized for 'Other' category.

4.6.2 Retrieve the related data e-mail data set for the classification.

Required data can be retrieved from bank authenticated mail system. They can be insert into a Microsoft excel worksheet in order to classify effectively.

4.6.3 Data Cleaning and pre - processing

The e-mail data is using the casual language and other symbols in the text. Data in the real world is dirty. They are incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data and noisy: containing errors or outliers inconsistent: containing discrepancies in codes or names. We can do data cleaning by filling in missing values, smoothing noisy data, identifying and remove outliers, and resolving inconsistencies. Data cleaning and pre- processing methodologies can be adopted to the system as follows.

4.6.3.1 Stop words

When combined with text classification applications, we often hear "empty" or "empty list" and even "stop list." Basically, a common set of empty words in any language, not just English. Why talk is essential for many applications on the grounds that, if we get rid of those frequently used in a particular language, we can focus on the important words instead. For example, in the case of a search engine if the search query is "how to develop information retrieval application" if the search engine tries to find pages containing the word "how", "should", "development", "Info", "restore" "applications", the search engine will find more pages that contain the word "how", "to" contains information about application development recovery information page information, because the term "how" and "who" are generally in English language in. So, if we ignore these two words, the search engines can really focus on the recovery of the page containing the keyword: "development", "Info", "restore", "application" - that will bring more than close the page is real interest. In order to sort e-mail efficiently. [15] .

We have remove 'stop words' contain in the e-mails to effective classification. English words commonly categorized as 'stop words' are freely available and they can be retrieved used in algorithm used to classification. By using 'stop words' list we can remove or avoid them while classification.

4.6.3.2 Pre - Processing

Before entering the relevant e-mails to the database through the classification algorithm those e-mails need to pre - process by removing following contents

- Remove or avoid meaningless words such as singhala words which could not provide any English meaning.
- Customer sensitive data such as NIC number, customer name , communication address etc.

The repository of meaningful data which available is used in this algorithm to avoid meaningless data and customer sensitive data. By using that repository of words developed algorithm can get only needed words for the classification and to train the algorithm.

I have adopted data cleaning and pre - processing mechanism which is explain above. Therefore I would be able to feed the developed algorithm with meaningful data. Form this method algorithm training and classification will be more effective. Because before filling the data to the algorithm they are cleaning and pre- processing for meaningful words as much as possible.

4.6.4 Develop classification using Bayes' Theorem

4.6.4.1 Bayes' Theorem

Definition : A theorem about conditional probabilities: the probability that an event A occurs given that another event B has already occurred is equal to the probability that the event B occurs given that A has already occurred multiplied by the probability of occurrence of event A and divided by the probability of occurrence of event B.[16]

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

Equation 1

4.6.4.2 Train E-mail classification algorithm using Bayes' Theorem

- Classify first mail in the dataset in to a one category (Request, Complain or Other)
- Feed it to the algorithm from the system.
- Calculate positive probability for each words in the e-mail for three categories.

Example :

- Customer e-mail received : (this e-mail is categorized as a 'Request' mail)

"Dear Sir, Due to urgent financial requirement kindly cancel the above FD immediately. Thank you "

- Keywords after cleaning and pre - processing

"kindly","dear"," due"," immediately"," close"," cancel"," financial"," requirement","urgent"

above all keywords are in 'Request e-mail received to the bank authenticated mail system. So positive count for the 'R' will be one and negative count for other two 'C' and 'O' will be individually one as follows

Word	Type	Positive Count	Negative Count
cancel	C		1
cancel	O		1
cancel	R	1	
close	C		1
close	O		1
close	R	1	
dear	O		1
dear	C		1
dear	R	1	
due	O		1
due	C		1

due	R	1	
financial	O		1
financial	C		1
financial	R	1	
immediately	R	1	
immediately	O		1
immediately	C		1
kindly	O		1
kindly	R	1	
kindly	C		1
requirement	R	1	
requirement	C		1
requirement	O		1
urgent	O		1
urgent	R	1	
urgent	C		1



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

- Then calculate positive probability of each word can be occur in each category using Bayes' Theorem as follows

$$\text{Positive probability} = \frac{\text{Positive count}}{(\text{Positive word count} + \text{Negative Count})}$$

Word	Type	Positive Count	Negative Count	Positive Probability
cancel	C		1	0.01
cancel	O		1	0.01
cancel	R	1		0.99
close	C		1	0.01
close	O		1	0.01

close	R	1		0.99
dear	O		1	0.01
dear	C		1	0.01
dear	R	1		0.99
due	O		1	0.01
due	C		1	0.01
due	R	1		0.99
financial	O		1	0.01
financial	C		1	0.01
financial	R	1		0.99
immediately	R	1		0.99
immediately	O		1	0.01
immediately	C		1	0.01
kindly	O		1	0.01
kindly	R	1		0.99
kindly	C		1	0.01
requirement	R	1		0.99
requirement	C		1	0.01
requirement	O		1	0.01
urgent	O		1	0.01
urgent	R	1		0.99
urgent	C		1	0.01

Table 2

4.6.4.3 Get probability of new e-mail from train algorithm.

To get probability for new entered email Category I have used Bayes' Theorem with multiple random variables

$$\begin{aligned}
P(A|X_1, X_2, X_3, \dots, X_n) &= \frac{P(A)P(X_1, X_2, X_3, \dots, X_n|A)}{P(X_1, X_2, X_3, \dots, X_n)} \\
&= \frac{P(A) \prod_{i=1}^n P(X_i|A)}{P(A) \prod_{i=1}^n P(X_i|A) + P(\neg A) \prod_{i=1}^n P(X_i|\neg A)}
\end{aligned}$$

Equation 2

System calculate positive probability of each key word of entered E-mail. It calculates the probability for each category 'request', 'complain' and 'other' by using Bayes' Theorem. After calculating probability for each category it decides e-mail related category from highest probability value. Furthermore system train the developed algorithm from new e-mail. Because of self - training feature of the algorithm

4.7 Summary

This chapter described and elaborate about methodology and classification going to used in this project and next chapter system design will be done based on described methodology.



University of Moratuwa, Sri Lanka.
 Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Chapter 5

Systems Design

5.1 Introduction

This chapter will first describe the functional view of the proposed system and the Architectural view. Then it will further describe the methodology used for the system development.

5.2 Architectural Design of the system

The proposed system is a classification system design using text classification techniques. By using text classification model bank users are able to classify the e-mails received from the customer, find synonyms and similar words using word net database.

Proposed system is in two layer architecture, "*Pre - processing layer*" and the "*Classification layer*". Then second layer classification layer or the data modelling layer is containing system logic/ algorithms for document classification and grouped set of synonyms (synsets).



University of Moratuwa, Sri Lanka
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

The pre-processing layer will contain components need to retrieve e-mails and data cleaning and pre-processing . From the techniques and methodology described earlier chapter 4 system will be designed for the text classification.

In pre-processing layer it contains following sub segments

- Component for read the e-mails retrieved from bank authentication system to Microsoft excel document.
- Component for remove stop words.
- component for data pre - processing (remove meaningless words and customer sensitive data)
- Component for enter new e-mail to get the category and change the category if required.

In Classification layer components contains following sub segments

- Use WordNet for avoid repeating words with similar meaning.
- Insert processed data to Classification DB.

- Calculate positive probability using Bayes' Theorem.
- Insert Training data to classification DB.
- Classify new e-mail using trained algorithm and self train the algorithm from provided e-mail.

5.3 Pre - processing layer

Pre- processing layer mainly design for the increase the correctness of the data which will be inputting to the system. The accuracy of the pre - processing layer will be discussed in the evaluation chapter.

The component for read E-mail need to be developed to read the provided data set from the provided format. In this case data will be provided in the Microsoft Excel Worksheet. System will be reading the Category, Subject and body content of the E-mails of the data set. To design this E-mail reading component I used sequence reading method in Java ('for loop').

Once data set is entered to the system sequentially system should be able to start the pre-processing by using the data feed. Removing 'Stop Words' are net level of the system. System will remove stop words from the received data set. To accomplish this task I have adapted the technique of reading 'Stop Word' list from the given text document and If any of those words exist in the data set system will replace that word from null. Therefore those words will not be count for the classification or train the algorithm.

Next challenge to remove customer sensitive data from the System and retrieve the words which will provide positive impact to classification criteria. For this also I have adapted the text document reading method. The meaningful words are saved to the words document and once data set is input to the system, system will get only the words in the words in the text document as the key words. This will remove customer sensitive data and meaningless data from the data set.

Last component of the classification layer allows user to enter new E-mails to receive its category. Once new E-mail is entered to the system will send that E-mail to the 'Classification Layer' to classify and then the according to result received from the classification layer it provide category of the new E- mail as the output. Furthermore it allows user to change the E- mail category if required.

5.3.1 Architectural Design of the pre - processing layer

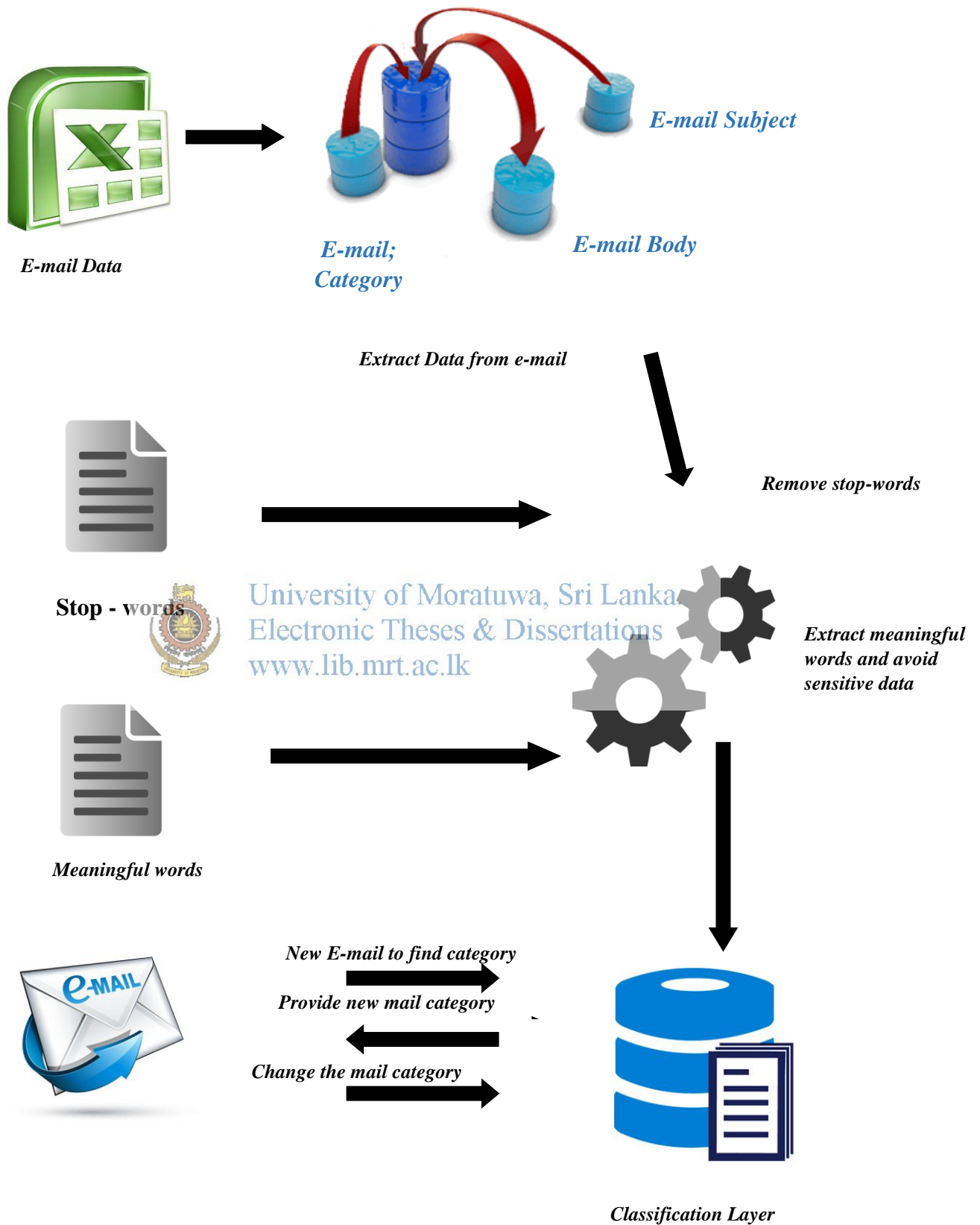


Figure 2

5.4 Classification layer

Classification layer is the vital component of the E-mail classification system. Once pre processed data set received from the pre-processing layer. System design to Use WordNet for avoids repeating words with similar meaning. This will act vital role in my system to increase the accuracy of the algorithm developed. Once pre-processed data received to the system each word check synonyms from WordNet database. Then all the synonyms received from the WordNet will be passing to the database. If any of those words are exist in the database system will increase positive count of that word and do not put new entry to the database.

Then system should insert data to database and calculate positive probability. At the time of training the algorithm positive probability of the algorithm will be calculating at the each time positive count or negative count increased in the word.

Finally trained algorithm from the data set will be able to classify entered new E-mail to the pre-defined category and self trained the algorithm from the new categorization.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

5.4.1 Architectural Design of the classification layer

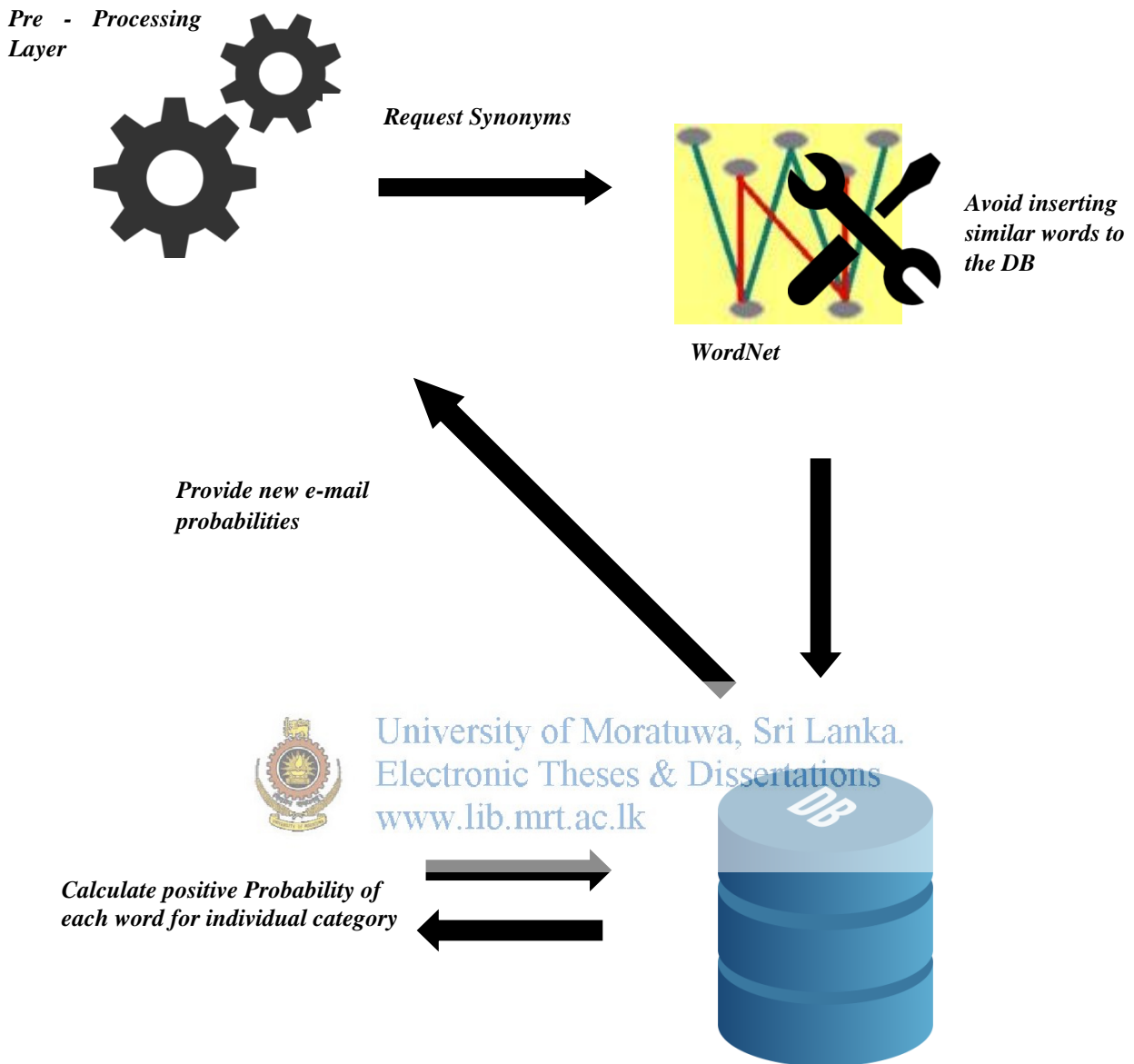


Figure 3

5.5 Summary

From the techniques described in earlier chapters above design can be derived and next chapter will describe about implementation of the conceptual design.

Implementation

6.1 Introduction

This chapter describes the algorithm development stage of the research. It discusses about implementation of pre - processing layer and classification layer.

6.2. Implementation of Pre - Processing Layer

The pre-processing layer will contain components need to retrieve e-mails and data cleaning and pre-processing . From the techniques and methodology describes earlier.

6.2.1 Component for read the e-mails retrieved from bank authentication system to Microsoft excel document.

- System allows to enter number of rows in a excel sheet and it name with the location.
- Then its reads that number of rows from the excel sheet.
- After reading the excel programme will submit the data to data cleaning and pre - processing criteria.

```
for (int i = 1; i < sheet.getRows() && i <= 1; i++) {  
  
    cell = sheet.getCell(0, i); //For type  
  
    type = cell.getContents() != null ? cell.getContents() : "";  
  
    System.out.println("Type of Meassage No " + (i - 1) + " Removed Common  
Words : " + type);  
  
    cell = sheet.getCell(1, i); //For Subject  
  
    subject = removeCommonWords(cell.getContents().replaceAll("(\\r\\n)", "  
").replaceAll("[^A-Za-z ]", "").replaceAll("\\s+", " ").trim(), commonWords,  
englishWords);  
  
    System.out.println("Subject of Meassage No " + (i - 1) + " Removed  
Common Words : " + subject);  
  
    co.trainingFilter(subject, type, database);  
  
    cell = sheet.getCell(2, i); //For Content
```



```

        content = removeCommonWords(cell.getContents().replaceAll("\\r\\n", "
").replaceAll("[^A-Za-z ]", "").replaceAll("\\s+", " ").trim(), commonWords,
englishWords);

        System.out.println("Content of Meassage No " + (i - 1) + " Removed
Common Words :" + content);

        co.trainingFilter(content, type, database);

    }

```

6.2.2 Component for remove stop words.


After reading the dataset from the provided excel sheet system will remove stop - words.

- From following component system gets provided stop - words and return for stop words removal

```

public Set<String> getCommonWords() {

    Set<String> stopWords = new LinkedHashSet<String>();

    try {
        
        BufferedReader br = new BufferedReader(new
        FileReader("src/resources/stopwords-list.txt"));

        String words = null;

        while ((words = br.readLine()) != null) {

            stopWords.add(words.toLowerCase().trim());

        } br.close();

    } catch (Exception ex) {

        ex.printStackTrace();    }    return stopWords;

}

```

- From following component system will remove the stop - words

```

public String removeCommonWords(String string, Set<String> commonWords,
Set<String> englishWords) {

    StringBuilder sb = new StringBuilder();

```

```

try {
    String[] tokens = string.split(" ");
    for (String token : tokens) {
        if (!commonWords.contains(token.trim().toLowerCase()) &&
            englishWords.contains(token.trim().toLowerCase())) {
            sb.append(token.toLowerCase().trim());
            sb.append(" ");
        }
    }
} catch (Exception e) {
    e.printStackTrace();
}
return sb.toString();
}

```

6.2.3 component for data pre - processing

To remove meaningless words and customer sensitive data. Following component will provides meaningful words which are affecting the classification. Using following component we can remove meaningless words and customer sensitive data.

```

public Set<String> getEnglishWords() {
    Set<String> englishWords = new LinkedHashSet<String>();
    try{BufferedReader
br = new BufferedReader(new FileReader("src/resources/english-words.txt"));
    String words = null;
    while ((words = br.readLine()) != null) {
        englishWords.add(words.toLowerCase().trim());
    }
    br.close();
} catch (Exception ex) {
    ex.printStackTrace();
}
return englishWords
}

```

6.2.4 Component for enter new e-mail to get the category and change the category if required.

Following component will allow you to enter new e-mail to the system

```
String result = "";

String msg = " new e- mail";

result = co.getResult(msg, commonWords, englishWords, database);

if(result.equals("C")){

    System.out.println(" Tested Messsage type : Complain");

}else if(result.equals("R")){

    System.out.println(" Tested Messsage type : Request");

}else if(result.equals("O")){

    System.out.println(" Tested Messsage type : Other");    }

}
```

Following three components will allow you to change the type as per the requirement

As mentioned above algorithm will trained according to the new user decision

- Change new e-mail type to 'Complain' mail and train the algorithm

```
co.changeMsgType(msg,ApplicationConstants.COMPLAIN_TYPE,
ApplicationConstants.REQUEST_TYPE,commonWords, englishWords,database);
```

- Change new e-mail type to 'Request' mail and train the algorithm

```
co.changeMsgType(msg, ApplicationConstants.REQUEST_TYPE,
ApplicationConstants.COMPLAIN_TYPE, commonWords, englishWords,database);
```

- Change new e-mail type to 'Other' mail and train the algorithm

```
co.changeMsgType(msg, ApplicationConstants.Other_TYPE,
ApplicationConstants.OTHER_TYPE, commonWords, englishWords,database);
```

- Train the algorithm according to change category

```
private void updateWordWithChangeType(String word, String type, int msgStatus) {

    Connection con = null;
```

```

PreparedStatement ps = null;

StringBuilder sbQuery = new StringBuilder();

Connections connections = new Connections();

int status = 0;

try {

    con = connections.getConnection(ApplicationConstants.EMAIL_DB);

    sbQuery = new StringBuilder();

    if (msgStatus == ApplicationConstants.POSITIVE_MSG) {

        sbQuery.append("UPDATE EMAIL_WORDS_TABLE SET
POSITIVE_COUNT = (NVL(POSITIVE_COUNT, 0) + 1), NEGATIVE_COUNT =
CASE WHEN (NVL(NEGATIVE_COUNT, 0) - 1) < 0 THEN 0 ELSE
(NVL(NEGATIVE_COUNT, 0) - 1) END ");

        sbQuery.append("WHERE WORD =");

        sbQuery.append(word);

        sbQuery.append(" AND TYPE = ");
        sbQuery.append(type);
        sbQuery.append("");

    } else {

        sbQuery.append("UPDATE EMAIL_WORDS_TABLE SET
POSITIVE_COUNT = CASE WHEN (NVL(POSITIVE_COUNT, 0) - 1) < 0 THEN 0
ELSE (NVL(POSITIVE_COUNT, 0) - 1) END, NEGATIVE_COUNT =
(NVL(NEGATIVE_COUNT, 0) + 1) ");

        sbQuery.append("WHERE WORD =");

        sbQuery.append(word);

        sbQuery.append(" AND TYPE = ");

        sbQuery.append(type);

        sbQuery.append("");

    }

    //System.out.println("CommonOperations > updateWord >> Update Sql :
"+sbQuery.toString());

```



```

        ps = con.prepareStatement(sbQuery.toString());

        status = ps.executeUpdate();

    } catch (Exception e) {

        System.err.println("CommonOperations > updateWord Error : " +
e.getMessage());

    } finally {

        connections.closeConnection(con);

    }

}

```

6.3 Implementation of Classification Layer

The second layer classification layer or the data modelling layer is containing system logic/ algorithms for document classification and grouped set of synonyms (synsets). It contain following components

6.3.1 Use WordNet for avoid repeating words with similar meaning.

This component will help main component to remove words with similar meanings.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

```

public String getSynonymsForSQL(Synset[] synsets) {

    StringBuilder sb = new StringBuilder();

    try { if (synsets.length > 0) {

        ArrayList<String> al = new ArrayList<String>();

        // add elements to al, including duplicates

        HashSet hs = new HashSet();

        for (int i = 0; i < synsets.length; i++) {

            String[] wordForms = synsets[i].getWordForms();

            for (int j = 0; j < wordForms.length; j++) {

                al.add(wordForms[j].replaceAll("'", "").toLowerCase());

            } }

        //removing duplicates

        hs.addAll(al);

```

```

al.clear();

al.addAll(hs);

//showing all synsets

for (int i = 0; i < al.size(); i++) {

    if(i == 0){

        sb.append("(");

    } sb.append(" ");

    sb.append((String)al.get(i));

    sb.append(" ");

    if(i == al.size()-1){

        sb.append(")");

    }else{

        sb.append(",");

    }

} catch (Exception e) {

    System.err.println("CommonOperations > getSynonymsForSQL Error : " +
e.getMessage());    }

return sb.toString();

```



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

6.3.2 Train the algorithm with pre - processed data and Calculate positive probability using Bayes' Theorem.

Following components will train the algorithm with pre - processed data and using developed logic using bayes' theorem.

```

private void updateWord(String word, String type, int msgStatus) {

    Connection con = null;

    PreparedStatement ps = null;

    StringBuilder sbQuery = new StringBuilder();

    Connections connections = new Connections();

    int status = 0;

```

```

try {
    con = connections.getConnection(ApplicationConstants.EMAIL_DB);
    try {
        if (msgStatus == ApplicationConstants.POSITIVE_MSG) {
            sbQuery.append("INSERT INTO EMAIL_WORDS_TABLE (WORD,
TYPE, POSITIVE_COUNT) ");
            sbQuery.append("VALUES (");
            sbQuery.append(word);
            sbQuery.append(", ");
            sbQuery.append(type);
            sbQuery.append(", 1) ");
        } else {
            sbQuery.append("INSERT INTO EMAIL_WORDS_TABLE (WORD,
TYPE, NEGATIVE_COUNT) ");
            sbQuery.append("VALUES (");
            sbQuery.append(word);
            sbQuery.append(", ");
            sbQuery.append(type);
            sbQuery.append(", 1) ");
        }
        //System.out.println("CommonOperations > updateWord >> Insert Sql :
"+sbQuery.toString());
        ps = con.prepareStatement(sbQuery.toString());
        status = ps.executeUpdate();
    } catch (SQLException e) {
        if (e.getMessage().startsWith("ORA-00001")) {
            sbQuery = new StringBuilder();
            if (msgStatus == ApplicationConstants.POSITIVE_MSG) {

```



```

        sbQuery.append("UPDATE EMAIL_WORDS_TABLE SET
POSITIVE_COUNT = NVL(POSITIVE_COUNT, 0) + 1 ");

        sbQuery.append("WHERE WORD =");

        sbQuery.append(word);

        sbQuery.append(" AND TYPE = ");

        sbQuery.append(type);

        sbQuery.append("");

    } else {

        sbQuery.append("UPDATE EMAIL_WORDS_TABLE SET
NEGATIVE_COUNT = NVL(NEGATIVE_COUNT, 0) + 1 ");

        sbQuery.append("WHERE WORD =");

        sbQuery.append(word);

        sbQuery.append(" AND TYPE = ");

        sbQuery.append(type);
        sbQuery.append("");
    }

    //System.out.println("CommonOperations > updateWord >> Update Sql
: "+sbQuery.toString());

    ps = con.prepareStatement(sbQuery.toString());

    status = ps.executeUpdate();

    }

    }

} catch (Exception e) {

    System.err.println("CommonOperations > updateWord Error : " +
e.getMessage());

    } finally {

        connections.closeConnection(con);

    }

}

```



6.3.3 Classify new e-mail using trained algorithm and self train the algorithm from provided e-mail.

```
String result = "";

String msg = "as i request it is greatly appreciated if you could kindly consider
sending statements by email.\n" +

"\n" +

"may i know the amount over due in my credit card and also the difference between
CLOSING BALANCE AND CURRENT BALANCE please.\n" +

"\n" + "\n" + "\n" + "brgds\n" + "\n" + "hiran\n" + "\n" + "Mr B V D H
Sagarachandra\n" + "No;127a\n"

"Walapala\n" + "Panadura";

result = co.getResult(msg, commonWords, englishWords, database);

if(result.equals("C")){

    System.out.println(" Tested Messsage type : Complain");

}else if(result.equals("R")){

    System.out.println(" Tested Message type : Request");

}else if(result.equals("O")){

    System.out.println(" Tested Messsage type : Other");

}

}
```

```
public String getResult(String msg, Set<String> commonWords, Set<String>
englishWords, WordNetDatabase database) {

String msgType = "";

try {      msg = removeCommonWords(msg, commonWords, englishWords);

double Cprob = getTypeProbability(msg,
ApplicationConstants.COMPLAIN_TYPE,database);

double Rprob = getTypeProbability(msg,
ApplicationConstants.REQUEST_TYPE,database);

double Oprob = getTypeProbability(msg,
ApplicationConstants.OTHER_TYPE,database);

}
```

```

Map hmSorter = new TreeMap();

hmSorter.put(Cprob, "C");

hmSorter.put(Rprob, "R");

hmSorter.put(Oprob, "O");

Iterator it = hmSorter.entrySet().iterator();

while (it.hasNext()) {

    Map.Entry pairs = (Map.Entry) it.next();

    System.out.println("FROM LOOP " + pairs.getKey() + " = " +
pairs.getValue());

    msgType = (String) pairs.getValue();

} //tra According to the result given by the algorithm

trainingFilter(msg, msgType, database);

} catch (Exception e) {

    e.printStackTrace();

}

return msgType;

}

```



6.4 Summary

System developed using Object Oriented Programming methodology and Java programming language. This chapter briefly described about implementation of each component. The evaluation of the developed system using technologies and classification techniques is describing in the next chapter.

Evaluation

7.1 Introduction

This chapter describes the evaluation of the research and the discussion. This chapter also includes the improvements and limitations of the methods used.

7.2 Evaluation

Several evaluation methods were used to ensure that the system meets its user requirements. Mainly used evaluation methods in testing phase are component testing, integration testing. Other than standard testing procedures I have adapted techniques to evaluate accuracy and effectiveness of the algorithm.

7.2.1 Component testing

The provided system has following components

- Read the E-mails
- Remove stop words
- Data pre - processing
- WordNet for avoid repeating words
- Calculate positive probability
- Insert Training data to classification DB
- Classify new e-mail using trained algorithm
- Self train the classification algorithm

7.2.1.1 Evaluate 'Read E-mails' Component

The developed component will read emails from the Microsoft excel worksheet. This was very effective comparing to read separate text documents. It provides unique advantage to separately read subject, body and the pre-defined category of each E-mails. Referring to the complex operations need to be done after data read 'Read E-mails' should be a efficient operation. Using excel data sheet was provided opportunity to achieved that objective. Also that component read each E-mail correctly. Each row of the excel sheet is reading correctly and retrieve the E-mails correctly.

7.2.1.2 Evaluate ' Remove stop words' Component

This component is essential for increase the accuracy of the algorithm. So I have to find method for remove 'stop words' effectively. Furthermore removing stop words operation is a value addition for the system. So efficiency is a very important attribute for that component. That approach adapted for that is feed all stop words using text file to the algorithm to remove 'Stop Words' from the E-mails received

7.2.1.3 Evaluate ' Data pre - processing' Component

Data pre - processing component use for remove customer sensitive data and retrieve meaningful key words from the E-mail. The approach used for Remove stop words used for this component as well. Provided text document will feed the meaningful words to the algorithm and allows filter key words for classification. This text document includes nearly 350,000 words which can be used as key words for classification.

7.2.1.4 Evaluate ' WordNet' Component

The database contains 155,287 words organized in 117,659 synsets for a total of 206,941 word-sense pairs. WordNet distinguishes between nouns, verbs, adjectives and adverbs because they follow different grammatical rules. Every synset contains a group of synonymous words or collocations. Different senses of a word are in different synsets. In addition to the features mentioned, I also evaluated the performance of WordNet under different retrievals. (as applicable)

7.2.1.5 Evaluate ' Calculate Positive Probability' Component

Developed Component calculate positive probability using following logic.

$$\text{Positive probability} = \frac{\text{Positive count}}{(\text{Positive word count} + \text{Negative Count})}$$

Probability calculation is tested and occurring correctly. Probability number should greater than zero and less than one. However above mentioned algorithm can give results not in required range. As solution foe that concern I have introduced a logic which will apply 0.99, if positive probability is greater than 0.99 or else logic will apply 0.01, if positive probability is less than 0.01. Hence this will effectively increase the classification performance and accuracy of the algorithm.

7.2.1.6 Evaluate ' Classify new e-mail using trained algorithm' Component

This is one of the core components of developed in the algorithm. This will train the algorithm from given data set. Efficiency of whole algorithm will be depend on this component. Hence evaluation of this component will be described in upcoming place with more details.

7.2.1.7 Evaluate Self train the classification algorithm Component

After training the algorithm this will provide correct category for new E-mail. Efficiency of whole algorithm will be depend on this component. Hence evaluation of this component also will be described in upcoming place with more details.

7.2.2 Integration Testing

Each individual component was integrated to according to the design which was described in Systems Design chapter. each component was integrated and tested for performance and accuracy. Integration testing was the most important testing stage when its look in to system as a whole.

7.2.2.1 Implementation of Integration testing

I have implemented Integration testing as described in following details

- Test scenario one : Test Read E-mail from the excel work sheet
 - System should read the Individual E-mail from provided Microsoft excel work sheet.
 - The E-mails data read from the system should include pre-defined subject from the user, E-mail subject and the E-mail body.
- Test scenario two : Test the algorithm training
 - System should remove the stop words from the read E-mail.
 - System should ignore customer sensitive data from the E-mail
 - System should ignore the words cannot provide a meaning.
 - System should provide synonyms from 'WordNet' database.
 - System should check the existence of each word in database.
 - If any word from the words set received from 'WordNet' exist in the database, the system should update the positive count of existing word.
 - System should calculate the positive probability of each word in the each category.

- Test scenario Three: Test the new E-mail classification.
 - System should calculate probability of each category for new E-mail.
 - System should view probable category based on the highest calculated probability.
 - System should increase the positive count and negative count based on given category to self train the algorithm.
 - System should allow changing the provided category from the classification algorithm.
 - System should increase the positive count and negative count based on changed category to self train the algorithm.

7.3 Summary

Testing is a vital stage of the Software Development Life cycle. Therefore I had focused seriously for the system testing for individual components and system as a whole. Next chapter will conclude the research.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Conclusion

8.1 Introduction

This chapter also includes the improvements and limitations of the methods used.

8.2 Analysis of Performance and Accuracy.

The performance and accuracy of the developed algorithm depends on following two components

1. Algorithm Training component which includes pre - processing layer and probability calculation component of the Classification layer.
2. New E-mail classification component which includes self training component as well.

8.2.1 Algorithm training performance

I have feed the algorithm by different number of E-mails with a pattern and get the time taken to complete the task.

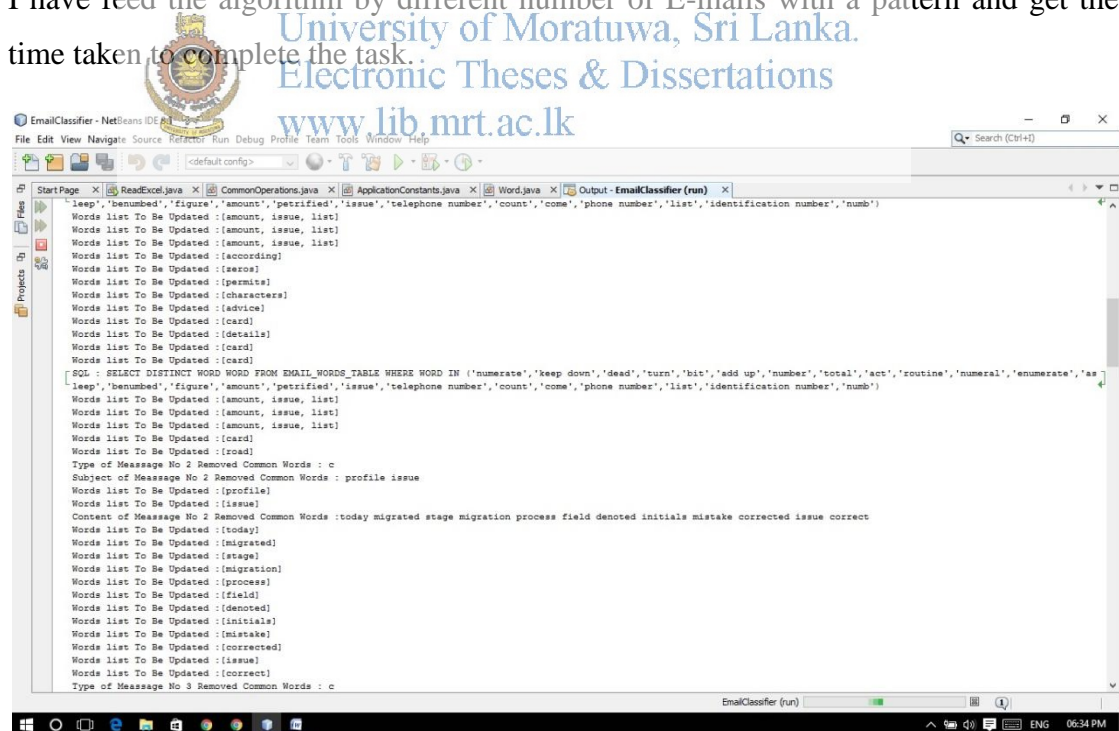


Figure 4 Training The E-mail Classification System from data set

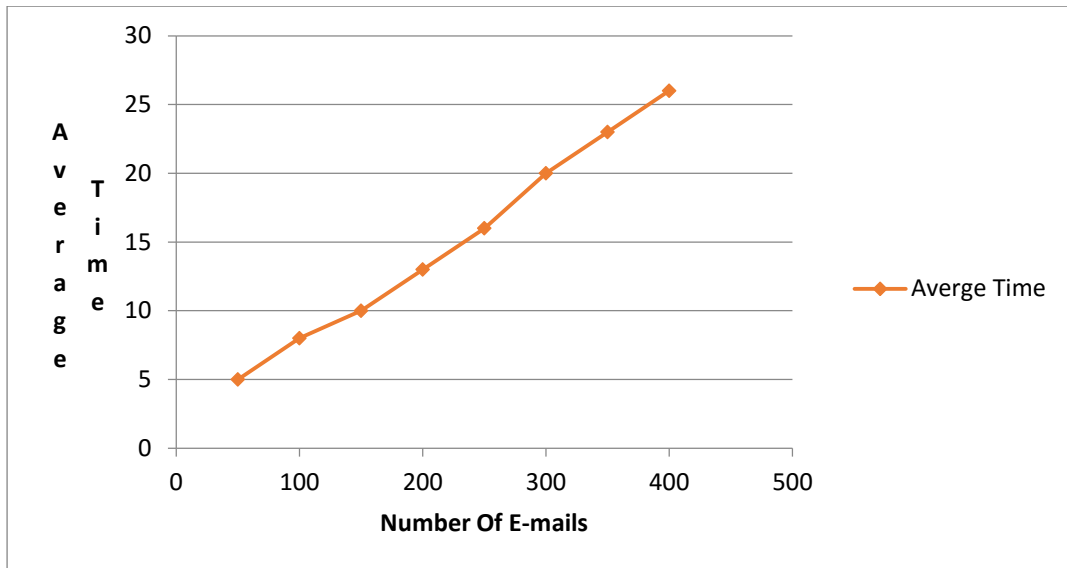


Table 3

8.3 Algorithm drawbacks

This algorithm ignores words where users have typed Sinhala words in English, for the algorithm it doesn't become much effective so using Wordnet and data pre-processing those singhala words are simply ignored. Because it simply sees the meaning of the word and synonym of that word, also the bag of words of each of the categories doesn't include Sinhala words in English thus this algorithm simply ignores them.

8.4 Limitations

- The extensive use of natural language, where a small number of users type Sinhala words in English, for the algorithm it doesn't become much of an issue but for use of wordnet the word is simply ignored.
- Meaningless e-mails, questions and text copied to give a output about some other thing causes trouble because the words are misleading and cannot be defined even by the expert.

8.5 Future Work

In the future, I would add following features to the algorithm

- All the weight of each term in the documents and pruning the terms with lower weight. This will increase the performance of the algorithm because low number of key words to search as well as reduce the garbage data for some extent, thus accuracy of the algorithm will be increased.

- Improve our simple classifier by using confidence based algorithm.[17]

8.6 Summary

After described all work I had done for successfully developed '**E-mail classification System for Bank Internal mail System**' final chapter described about the performance analysis of the solution given by employing different parameters and algorithms and draw backs. Finally future work also mention in this chapter.



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

References

- [1] “What is EDI (Electronic Data Interchange)?,” *EDI Basics*. [Online]. Available: <http://www.edibasics.com/what-is-edi/>. [Accessed: 07-Mar-2016].
- [2] “Complaint Handling - Sampath Bank PLC.” [Online]. Available: <http://www.sampath.lk/en/contact/complaint-handling>. [Accessed: 07-Mar-2016].
- [3] M. K. Dalal and M. A. Zaveri, “Automatic text classification: a technical review,” *Int. J. Comput. Appl.*, vol. 28, no. 2, pp. 37–40, 2011.
- [4] T. P. Jurka, L. Collingwood, A. E. Boydston, E. Grossman, and W. van Atteveldt, “RTextTools: A supervised learning package for text classification,” *R J.*, vol. 5, no. 1, pp. 6–12, 2013.
- [5] C. C. Aggarwal and C. Zhai, “A survey of text classification algorithms,” in *Mining text data*, Springer, 2012, pp. 163–222.
- [6] R. Kumar and R. Verma, “Classification algorithms for data mining: A survey,” *Int. J. Innov. Eng. Technol. IJIET*, vol. 1, no. 2, pp. 7–14, 2012.
- [7] S. I. Ao and International Association of Engineers, Eds., *International MultiConference of Engineers and Computer Scientists, IMECS 2009:: 18 - 20 March, 2009, Regal Kowloon Hotel, Kowloon, Hong Kong*. IAENG, 2009.
- [8] V. C. Gandhi and J. A. Prajapati, “Review on Comparison between Text Classification Algorithms.”
- [9] U. Fayyad, “Data mining and knowledge discovery in databases: implications for scientific databases,” in *Scientific and Statistical Database Management, 1997. Proceedings., Ninth International Conference on*, 1997, pp. 2–11.
- [10] T. N. Mansury and R. J. Hilderman, “Evaluating WordNet Features in Text Classification Models,” in *FLAIRS Conference*, 2006, pp. 568–573.
- [11] Z. Elberichi, A. Rahmoun, and M. A. Bentaallah, “Using WordNet for Text Categorization,” *Int Arab J Inf Technol*, vol. 5, no. 1, pp. 16–24, 2008.
- [12] B. Klatt, K. Krogmann, and V. Kuttruff, “Developing Stop Word Lists for Natural Language Program Analysis,” *Proc. WSRE'14*, 2014.
- [13] H. Saif, M. Fernández, Y. He, and H. Alani, “On stopwords, filtering and data sparsity for sentiment analysis of Twitter,” 2014.
- [14] G. A. Miller, “WordNet: a lexical database for English,” *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [15] H. Saif, M. Fernández, Y. He, and H. Alani, “On stopwords, filtering and data sparsity for sentiment analysis of Twitter,” 2014.
- [16] “Bayes’ theorem | Define Bayes’ theorem at Dictionary.com.” [Online]. Available: <http://dictionary.reference.com/browse/bayes--theorem>. [Accessed: 05-Mar-2016].
- [17] M. Dredze, K. Crammer, and F. Pereira, “Confidence-weighted linear classification,” in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 264–271.