IMPROVING CUSTOMER SERVICE USING PUBLIC OPINION MINING

A Thesis Presented

by

Kalkidan Mekonnen

to

The Faculty of Informatics

of

St. Mary's University

In Partial Fulfillment of the Requirements
For the Degree of Master of Science
In
Computer Science

February, 2022

ACCEPTANCE

Improving Customer Service Using Public Opinion Mining

By

Kalkidan Mekonnen

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

____

Internal Examiner

____

External Examiner

____

Dean, Faculty of Informatics

February 2022

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

Kalkidan Mekonnen Asmamaw

____

____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

Dr. Getahun Semeon, Ph.D.

____

____

Signature

Addis Ababa

Ethiopia

February, 2022

ACKNOWLEDGMENT

I am highly indebted to all people who helped me to finalize this paper. The names to be enumerated in this regard are many and include those who have given me their heartfelt advice and those who took their precious time to complete and prepared. My special thanks also go to my advisor, Getahun Semeon (PHD), I am extremely grateful for your unreserved and valuable advice, constructive corrections, comments and suggestions and motivation throughout my thesis project.

Also I would like to extend my acknowledgement to Zerihun Tolla. Zersh, you are more than a friend. I am lucky to know you thank you for all your support. Thank you my friend Melese, for all your support and smooth friendliness. I would also like to thank my staff member and friend Selamawit for your support.

I owe my deepest gratitude to Nesredin and Henok for her encouragement and unreserved assistance. I will always cherish the time you spent with us.

Finally, I also like to express my appreciation to my family members who have helped me in so many ways.

List of figures

List of Tables

Abbreviations

APL − Application Programing Language

BAGGING −Bootstrap AGGregatING

HTML−Hyper Text Markup Language

NNET−Neural Network

SEMMA −Sample, Explore, Modify, Model and Assess

SGML−Standard Generalized Markup Language

SLDA -Supervised latent Dirichlet allocation

SVM – Support Vector Machine

Abstract

Today, digital reviews play a pivotal role in enhancing global communications among consumers and influencing consumer buying patterns. The availability of technology and infrastructure create opportunities for citizens to publicly voice their opinions over social media. Business Company uses this opportunity to improve the quality of their product and the efficiency of their company. Companies don’t yet have an effective way to make sense of customer opinions given on the product. Now a day’s huge amount of product reviews are posted on the Web. Such a product reviews are a very important source of information for business companies to know about their product acceptance by their customer. Manual analysis of these reviews is very difficult because of the increase in the numbers of reviews on products day after day. Techno Company creates a Facebook page which helps consumers to share their experience and provide real insights about the performance of the product to future buyers. In order to extract valuable insights from a large set of reviews, classification of reviews and rating products into 1for best product which is highly accepted by their customer, 2 for good product and 3 for products having problem which customers is not happy to buy it.is. Product review Analysis is a computational study to extract subjective information from the text.

This paper proposes a customer opinion analysis model to classify product reviews and rating the product best, good and bad based on the customer feedback about the product. It applies six popular machine learning classifiers namely: Support Vector Machine (SVM), BOOSTING, SLDA, NNETWOR, TREE and BAGGING with the aim to come up with the most efficient classifier. The dataset used consists of 2000 reviews about mobile phone products, collected from Tecno Facebook page. In order to evaluate the six classifiers, we used 10fold cross validation, recall, precision, F1-mesaure and accuracy to measure the performance of each algorithm. The results showed that SVM and BOOSTING outperformed the other classifiers in term of accuracy in all experiments. Decision Tree algorithm gave the lowest results across all experiments in terms of accuracy.

Keywords: opinions, Opinion Mining, Review, Sentence Level, Document Level, Feature Level, Classification, Extraction, Machine learning algorithms, Determination

Chapter One
Introduction

1.1. Background

Social media such as Facebook has become extremely popular these days. Millions of users use this platform to share their opinions, thoughts, and emotions. User-generated comments represent a potential complementary source of essential information about company’s brand that can be positive, neutral and negative [1]. The way companies and organizations improve their services are identifying and subsequently knowing consumer needs and their feelings. Opinion Mining or Sentiment Analysis is the evaluation model to learning of public opinions, attitudes and feelings toward any item, product or seller. The object can characterize persons, objects or topics so, in order to examine consumer needs and to implement effective marketing strategies aimed at satisfying their needs, marketing managers need relevant, current information about consumers, competitors and other forces in the market place [2]. As to the knowledge of the researchers there is no study conducted in Ethiopia on customer opinion mining that support developing and implementing effective marketing strategies based on the opinion mined. In the past as a significant part of consumer information, which is present on company’s website and social media pages, has been ignored. Web reviews can be used for this purpose as the web has acquired immense value as an actively evolving repository of knowledge for market research. [3][4]. Due to availability of large volume of information on Web, opinion mining can still be a challenging task [3] [10] [4].

Social media has an important impact on the field of business, advertisement, and e-commerce as it explains consumer behavior and feedback about particular business proposals, services and products. [1] Opinions and purchase decisions of the people and organizations are now affected and sometimes taken as a response to the content of social media before going to the market and actually test the product. In social media, all data from posts, comments and replies needs measuring results and concluding insights out of them rather than just reading the opinions of others, this is known as social media analytics. Social media analytics are the practice of gathering data from social media platforms and analyzing that data to make business decisions [2] [11]. The most common use of social media analytics is to mine customer sentiment in order to support marketing and customer service activities. The importance of social media analytics is intuitive and flexibly used by companies, organizations and individuals to know the insight of the market. It helps companies to know customers ‘viewpoints and their comments on the quality of the products and services to make successful business decisions. The typical objectives include increasing revenues, reducing customer service costs, getting feedback on products and services, as well as improving public opinion of a particular product or business division [7].

Opinion mining, which is also known as sentiment analysis, emotion mining, attitude mining or subjectivity mining [2] is a hot research discipline which is concerned with the computational study of opinions, sentiments and emotions expressed in an opinionated text.

This thesis aims at finding and classifying customer opinions given on Facebook page of Tecno mobile Product related to their product. Why is opinion mining important now? It is mainly because of the web, which is full of huge volume of opinionated text [5][6][8]. Sentiment mining can be done at sentence level, document level or feature level. In sentence level opinion mining, there are two tasks: the subjectivity classification and sentiment classification. The first is concerned with subjectivity and objectivity classification. Sentences are classified into pre-defined binary classification subjective sentence (e.g. it is such a nice phone) or objective sentence (e.g. I bought an iPhone a few days ago). The sentiment classification is concerned with polarity classification. The sentences are classified as positive (e.g. it is just a nice phone), negative (e.g. the phone broke in two days) or neutral classification. The document level sentiment classification is concerned with classifying the document based on the overall opinion expressed by the opinion holder as positive, negative and neutral. At the feature level sentiment mining, commented features are identified, extracted and the sentiment towards these features is determined [4] [8].

Analyzing these opinions is the main objective of opinion mining. The analysis of these opinions can be helpful in contexts as follows:

Production: here, opinion mining can be used to find defects in a product or aspects that are prone to be enhanced. For instance, a cell phone can be made with a sturdier material if users complain about its fragility.

Customer service: here, the satisfaction of users can be measured using their comments. For example, the selection of cell phone product offered to a mobile user can be improved if the battery, ram, camera and other accessories are inferred from their reviews about previous buyers.

Overall, the intent of the research project is to develop predictive model that takes Tecno mobile product social media user’s opinion on web and classifies the opinions as to use their feedback for improving customer service.

1.2. Statement of the Problem

Tecno Mobile used a social media platform dedicated to promoting and selling its products and services. One of social media account is Facebook, which is one of the most popular social media accounts, with more than 12,000,000 followers in Ethiopia alone. All Africa is twice as big commenting and promoting new products mainly in Nigeria, Uganda, Kenyan and South Africa. But this information has not been used in a modern way. Traditionally, Tecno mobile surveys are used to collect feedback from their customers in a structured manner. By compiling and disseminating queries, the data were collected and Analyze together as a percentage. It is difficult to get the required feedback as people are not interested in answering surveys; therefore fails to detect the most important problems. Due to the importance of internet and the ease of use any user, buyer or customer rely on the Web for their opinions on various cell phone products and services they have used, it is very important to develop methods to automatically classify and evaluate them.

In the past few years, a great attention has been received by web documents as a new source of individual opinions and experience. This situation is producing increasing interest in methods for automatically extracting and analyzing individual opinion from web documents such as customer reviews, weblogs and comments on news. Recently, electronic commerce websites use of the Internet has increased to the point that consumers rely on them for buying and selling [7]. Since these websites give consumers the ability to write comments about different products and services, huge amounts of reviews have become available [8]. Consequently, the need for to analyses those reviews to under-stand consumers' feedbacks have increased for both vendors and consumers. However, it is difficult to read the entire feedbacks for a particular item especially for the popular items with many comments [9]. Tecno Mobile product's used a social media platform dedicated to promoting and selling its products and services. To improve the quality of their product and add efficiency to their firm it is recommended to use internet user's reviews on the purchased product. Product reviews are a key in identifying the aspects of a product from which opinions originate and in establishing a comparison between products, product departments, and brands. In order for these comparisons to be valid, the sentiment analysis has to be executed over either the same product or a set of products in the same department, or a group of brands with some product departments in common. [10] [11]

Using this method is not efficient, as people are not interested in answering surveys therefore, fails to detect the most important problems. Therefore, it is important to develop a method to automatically classify and evaluate customer opinions on Tecno cell phone products and services they have used. The intent of this research is to build a predictive model for consumers' satisfaction on a Mobile phone product based on the reviews. We will also attempt to understand the factors that contribute to classify product reviews and rating the product best, good and bad based on the customer feedback about the product (based on important or most frequent words).the model helps a Tecno company to improve their product quality and service based on the customer opinion. The customer opinion is bad the company revises their weakness if the opinion is positive the company keeps their strength.

1.3. Research questions

The key questions included in this research are as follows.

1. How can Tecno company uses customer opinion to improve their product quality and service?
2. How can we analyze, classify customer feedback from opinion collected from web (social media) and evaluate using opinion mining method?

Considering the above research questions this research finds answers and builds a customer service analysis model to help the companies service by using customer opinion collected from web

1.4. Objectives

1.4.1. General objectives

The General objective of this research is to develop a model to predict user rating, usefulness of review by using opinion mining technique in order to help a Tecno company to improve their product quality and service

1.4.2. Specific objectives

In order to achieve the specific objective of this research the following task has been done.

- To explore application of data mining in opinion reviews on cellphone product.
- To explore classification algorithm on cellphone user's data.

- To identify, collect and organize resources that are necessary for building a customer opinion analysis model.
- To experiment and build a model that identifies efficiency of company.
- To construct a prototype and evaluate it.

1.5. Significance of Research

We develop a predictive model, which evaluates user’s satisfactions, and complain about the service of the organizations. It allows you to get inside your customers' heads and find out what they like and dislike, and why, so you can create products and services that meet their needs. The outcome of this research is expected to help many organizations in Ethiopian who gives services for customers to improve their efficiency and income. Thus, opinion mining plays a pivotal role to focus on the opinions which express or imply positive or negative sentiments. It can be applied to different products and services.

1.6. Scope and limitation of the Research

The scope of this research is limited to studying and building predictive models based on opinion and user’s feedback extracted from Tecno Mobile from social media. Also the model is limited to reviewing only opinions written in English language.

1.7. Thesis Organization

The reminder of this thesis is organized as follows. There are six sections. Section one of the research is introduction which explains about the research .In section two, under literature review background and other related researches conducted on opinion mining/sentiment mining be presented. In this part the concept of opinion mining and other related works is presented. In section three, methodology and other related procedure which helps to achieve the goal of project be presented. Following the methodology section four explains details of data collection, data preprocessing and datasets used in experiment to predict early threat. Under this section how to data are crawled and other multiple preprocessing tasks are described. Section five, discusses the development phase and discuss experiment and result. Finally, in Section six, conclusions and future work are provided.

Chapter Two
Literature Review

2.1. Overview

In the world of online shopping today, customers trust other shoppers more than they trust brands. That is why the most effective way to increase conversions and bring in new shoppers is to hand the microphone to those who already know and love your products and share their customer reviews with the world [15].

Government and companies do not yet have an effective way to make sense of this user's conversation and interact importantly with thousands of others. As a result, social media is characterized by short-termism and auto-preferentiality. Technology and infrastructure create opportunities for citizens to publicly voice their opinions over social media, but has created serious problems when it comes to making sense of these opinions. Many experts consider social media as opportunity for research. The concept of opinion mining comes after the use of social media increases and the amount of data on social media initiates the researcher on field of big data analytics [14] [15].

The new types of Internet content enforced new ways of data management which, as a consequence, caused new problems and opportunities to arise. Over the last decade a huge increase of interest in the sentiment analysis research is clearly visible [2]. Sentiment analysis on the opinion is about determining the subjectivity, polarity (positive or negative) and polarity strength (weakly positive, mildly positive, strongly positive, e.) of a piece of text in other words:

- ✓ What is the opinion of the writer?

In the other hands Opinion mining can be defined as a sub-discipline of computational linguistics that focuses on extracting people's opinion from the web [2][6][10]. The recent expansion of social media encourages users to contribute and express themselves via blogs, videos, social networking sites, etc. All these social media platforms provide a huge amount of valuable information in order to analyze. Opinion-mining systems analyze:

- ✓ Which part is opinion expressing;
- ✓ Who wrote the opinion;
- ✓ What is being commented

2.2. Opinion mining

Opinion Mining is the field to extract the opinionated text used different sources and summarized it in the understandable form for the end user. Solve problems related to the opinions about products, political leaders, ideas, and services [3]. A promising discipline which is defined as combination of information retrieval and computational linguistic techniques deals with the opinions expressed in a document. There are different techniques for summarizing customer reviews like Data Mining, Information Retrieval, Text Classification and Text Summarization [10]. Before World Wide Web users asked the opinions of his family and friends to purchase the product. In the very same way when any organizations need to take the decision about their products they had to conduct various surveys to the focused groups or they had to hire the external consultants to do so [3][4] [10]. Ease the customers to take decision to purchase the product by reviewing the posted comments. Customers can post reviews on web communities, discussion forums, blogs, product's web site these comments are called user generated contents. Web2.0 is playing a vital role in data extracting source in opinion mining. It facilitates users to know about the product from other customer's reviews that have already used it instead of asking friends and families. Companies, instead of conducting surveys and hiring the external consultants to know about the client's opinions, extract opinionated text from product web site [10]. Opinion mining is used to extract the positive, negative or neutral opinion summary from unstructured data. It involves subjectivity in text and computational management of opinion. It is the sub-discipline of web content mining, which involves Natural Language Processing and opinion extraction task to find out the polarity of any product consumers feedback [12]

2.2.1. Components of Opinion Mining

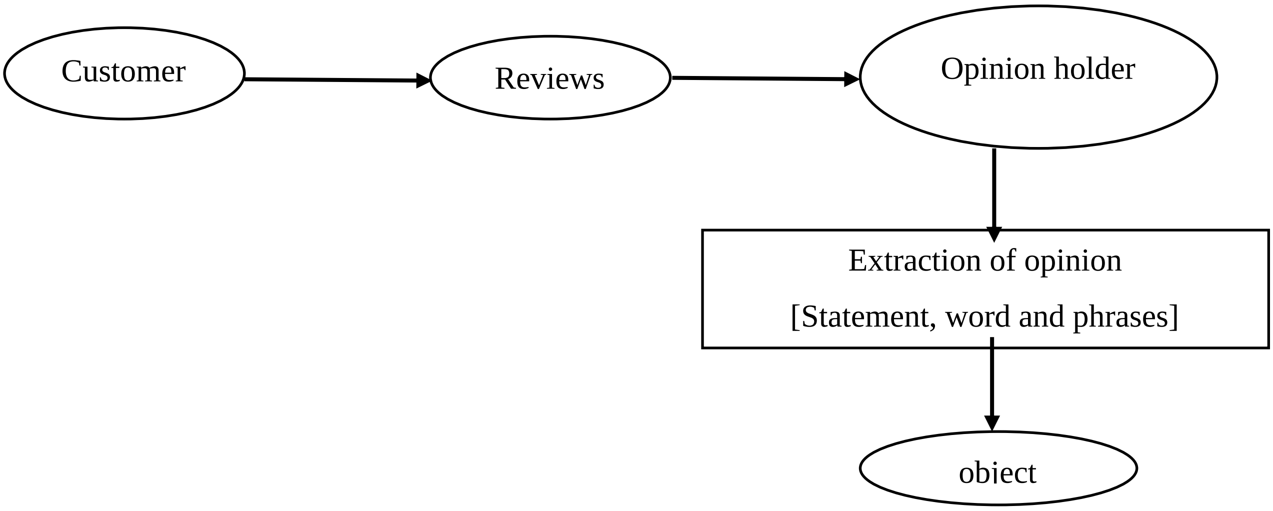


Figure 1 opinion mining model [10]

Figure 1 describes the object model of Opinion Mining there are five components i.e. a customer giving the broader reviews from various sources, it is the sentiment, views or judgment about any object based on knowledge or experience, then the opinion holder of a particular opinion; it may be a person or an organization that holds the opinion. In the case of blogs and reviews, opinion holders are those persons who write these reviews or blogs object which is an entity (person, topic, product or organization) about which the opinion expressed [10] [12].

2.2.2. Levels of Opinion Mining

Research in the field of sentient analysis is done at the following levels there are: -

- ✓ *Document level:* - the document level classifies the whole document as a single polarity positive, negative or neutral.
- ✓ *Sentence level:* - analyze the documents at sentence level. The sentences are analyzed individually and classified as positive, negative or neutral.
- ✓ *Aspect level:* - going much deeper and deals with identifying the features in a sentence for a given document and analyze the features and classify them accordingly as positive, negative or neutral [6] [13].

2.3. Sentiment classification

Sentiment classification is the automated process of identifying opinions in text and labeling them as positive, negative, or neutral, based on the emotions customers express within them. Using NLP to interpret subjective data, sentiment classification can help you understand how customers feel about your products, services, or brand [15] [16].

The aim of Sentiment classification is to classify tweets or sentences into three classes called positive, negative or neutral opinions. The process of sentiment classification includes training classifier, classify online tweets, and analyze classification results. In [16] a brief overview is given where three common sentiment analysis approaches are mentioned as Machine learning, lexicon-based methods and linguistic analysis.

Table 1 Sentiment classification approaches

Sentiment classification approaches		Features/techniques	Advantages and limitations
Machine learning	Bayesian,Netwoks , Naive Bayes, Classification Maximum Entropy, Neural Networks, Support Vector Machine	Term presence and frequency Part of speech information Negations Opinion words and phrases	ADVANTAGES the ability to adapt and create trained models for specific purposes and contexts LIMITATIONS the low applicability to new data because it is necessary the availability of labeled data that could be costly or even prohibitive
Lexicon based	Dictionary based approach Novel Machine Learning Approach Corpus based approach Ensemble approach	Manual construction, Corpus-based Dictionary based	ADVANTAGES wider term coverage LIMITATIONS finite number of words in the lexicons and the assignation of a fixed sentiment orientation and score to words
Hybrid	Machine learning Lexicon based	Sentiment lexicon constructed using public resources for initial sentiment detection Sentiment words as features in machine learning methods	ADVANTAGES lexicon/learning symbiosis, the detection and measurement of sentiment at the concept level and the lesser sensitivity to changes in topic domain LIMITATIONS noisy review

In this table machine learning classification technique to classify text consists of two sets documents training and a test set. The training set is used for learning the differentiating characteristics of a document. Test set is used for checking how well the classifier performs. The features of machine learning based for sentiment classification are-

- ✓ *Term presence and their frequency*: that includes Uni-grams or n-grams and their presence or frequency.
- ✓ *Part of speech information*: used for disambiguating sense which is used to guide feature selection.
- ✓ *Negations*: has the potential of reversing sentiments opinion words/phrases: that expresses positive or negative sentiments.

Machine learning filtering work is the following method. URL links and social media user names, then tokenizing the text message with punctuation marks and spaces, removing stop words, and constructing n-grams that is sets of n subsequent words. After extracting the text, Machine Learning methods can be used on the training data such as support vector machines, Random Forest, and Naïve Bayes. As [17] stated that Naïve Bayes works the best of the three mentioned methods. The reason why we use a Naïve Bayes classification algorithm is its computational and memory efficiency. It is also effective with small training data size and training time compared to the other methods, and it produces an oversimplified model with accurate classification performance.

The second approach is lexicon based a uses sentiment dictionary with opinion words and matches them with the data for determining polarity. There are three techniques to construct a sentiment lexicon: manual construction, corpus-based methods and dictionary based methods. The manual construction is a difficult and time-consuming task. Corpus-based methods can produce opinion words with relatively high accuracy.

Finally, hybrid approach or linguistic analysis approach. The combination of both the machine learning and the lexicon-based approaches has the potential to improve the sentiment classification performance. There are some advantages and limitations in using these different approaches depending on the purpose of the analysis. This analysis approach exploits the grammatical structure of the text in combination with a lexicon to predict its polarity. As a part of classification process Linguistic algorithms may attempt to identify context, negations, part-of-speech, i.e. noun, verb, etc. The main problem with linguistic analysis for the micro blog messages is the fact that most tweets are grammatically incorrect: having abbreviations, symbols, incomplete sentences due to the shortness of the written message [11] [17].

2.4. Text classification

The application of text classification is very important. In this era classification has always been an important application and research topic since the start of digital documents. Because of the availability of large amount of text documents that we have to work with

In This thesis, classify text as topic based and text genre based. The first type of text classification which is Topic-based text classification classifies documents according to their topics [12]. Texts can also be written in many categories such as: news, movie reviews, product reviews and advertisements. The second type of text categorization is defined on the way a text was created, the way it was edited, the language it uses, and the kind of people to whom it is addressed. Typically, most data for genre classification are collected from the web, through bulletin boards, news groups and printed news. The aim of Text Classification is to classify a document under a predefined category.

As in every supervised machine learning task, to classify the document the initial data set is needed. A document may be assigned to different category. The task of constructing a classifier for opinion extracted from social media does not differ a lot from other tasks of Machine Learning. The main issue is the representation of a document [13]. The major problem of tweet categorization is that the number of unique words or phrases which we use as a feature to classify document can easily reach orders of tens thousands. This raises complication in applying most machine learning algorithms to text classification. Thus we must use a lot of technique such as dimension reduction for better classification.

Two possibilities exist, either selecting a subset of the original features [14], or transforming the features into new ones, that is, computing new features as some functions of the old ones [15]. We examine this approach in this project. After selecting a best feature, a supervised machine learning algorithm can be applied. Some algorithms like support vector machine perform better opinion classification task and are more often used.

2.5. Machine learning algorithm

Machine learning algorithm used for the classification task is mainly classified into three categories.

Supervised

The supervised classifier if all the training datasets class for each input in the corpora is correctly labeled. Machine learning classifier method, which follows this approach, is summarized in the following table.

Semi-Supervised

Stated above supervised data depends on labeled data. That attribute limits their applicability. Machine Learning methods, which are able to combine, labeled with unlabeled data are called semi-supervised learning methods [24].

Unsupervised

When an analyst does not participate in the algorithm's learning process that is the classifier cannot use labeled training datasets we call such method as unsupervised classification technique. [17].

Many document classifiers have been proposed so far in the literature using machine learning techniques, probabilistic models, etc. They all differ in the method adopted: Naïve Bayes, decision trees, neural networks, nearest neighbors, rule induction, and lately, SVM. Even though many approaches have been proposed, the automated text classification is still a major area of research. This is mainly because the effectiveness of current automated text classifiers is not purely accurate and needs additional improvement.

Naive Bayes is often used in text classification applications and experiments because of its simplicity and effectiveness [24]. However, the classifying performance is often less because it does not model text classification well. Schneider addressed the problems and show that they can be solved by some simple corrections [18]. Klopotek and Which presented results of empirical evaluation of a Bayesian multinet classifier based on a new method of learning very large treelike Bayesian networks [14]. The study suggests by those researchers that tree-like Bayesian networks are able to handle a text classification task in 100,000 variables with sufficient accuracy and speed.

SVM, for text classification they provide poor recall but excellent precision. One way of customizing Support vector classifiers to improve recall, is to adjust the threshold associated with the classifier. An approach proposed by Shanahan and Roma, which clearly describe an automatic process for adjusting the thresholds of generic support vector classifier with better result [25].

Another researcher called Johnson et al. described a fast decision tree construction algorithm that takes advantage of the spirit of text data, and simplified rule approach that converts the decision tree into a logically equivalent rule set. Lim also proposed a good approach, which improves performance of k nearest neighbors, based text classification by using well-estimated parameters [24, 25]. Some variants of the k nearest neighbor approach with k values, different decision function and feature sets were proposed and evaluated to find out satisfactory parameters. Corner classification network is a kind of feed forward neural network for instantly text classification. A training algorithm, named as TextCC is presented in [26].

The degree of difficulty of document classification tasks obviously varies. As the number of distinct classes increases, so does the difficulty, and therefore the size of the training dataset needed. In any multi-class document classification task, without doubt some classes are more difficult than others to classify. Reasons for this classifier difference may be one of the following reasons:

1. very few positive training examples for the class
2. lack of good predictive features for that class

they are train a binary classifier per category in text categorization, use all the documents in the training corpus that belong to that category as relevant training data and all the documents in the training corpus that belong to all the other categories as non-relevant training data. It is often the case that there is an overwhelming number of non-relevant training documents especially when there is a large collection of categories with each assigned to a small number of documents, which is typically a problem of un balanced data. This problem indicates a particular challenge to classification algorithms, which can achieve high accuracy by simply classifying every example as negative. To overcome this problem, cost sensitive learning is needed [25].

A scalability analysis of a number of classifiers in text categorization is given in [24]. The author reviewed categorization experiments performed over noisy texts [20]. By noisy it is meant any text obtained through an extraction process from media than digital texts. For example, transcriptions of speech recordings extracted with a recognition system. Other authors [25, 26] also proposed to parallelize and distribute the process of text classification. With such a procedure, the performance of classifiers can be improved in both time complexity and accuracy.

The Algorithms and Classification Frameworks table shows the three frameworks and a list of machine-learning algorithms that can be applied to each one of them.

Table 2 List of machine learning algorithm and classification frame works

Algorithm	Supervised	Semi-Supervised	Unsupervised
Naïve Bayes	Yes	Yes	No
Maximum Entropy	Yes	Yes	No
K-Means Clustering	No	Yes	Yes
S.V.M.	Yes	Yes	No
Decision Tree	Yes	Yes	No
Bayesian Network	Yes	Yes	No
Artificial Neural Networks	Yes	Yes	Yes

2.6. Related work

In the field of more works on opinion, mining focused on the polarity of opinion, positive or negative; this kind of opinion mining is called sentiment analysis. Another type of opinion mining focused on finding the detailed information of a product from reviews; this approach is a kind of information extraction. Now much research has focused on assessing the review quality before mining the opinion. [5] Opinion mining, several efforts have been made to predict or classify threatening or offensive texts. Prediction of offensive text methods are described in [13] for detecting offensive languages in social media, using weak words and strong offensive words, combined with text mining techniques also used in sentiment analysis appraisal approach, like n-grams, Bag-of-Words. They also try to classify users as being offensive or not. The works listed below illustrate this: -

Since this work is interested in studying the sentiments of Mobile phones reviews on Amazon and Facebook the work related to analyzing the sentiments of Mobile phones or Amazon review shave been considered in the review. In the following, these researches are reviewed in terms of pre-processing techniques, feature extraction methods, proposed methodologies, and evaluation metrics. Various works in the literature have focused on the problem of identifying user's opinions of different products using Amazon reviews of "Unlocked Mobile Phones" [14] [15]. The work by [14] focused on a specific Brand Name, 'iPhone', to examine algorithms' validity in order to classify online reviews using a supervised model. On the other hand, [15] aggregated

40000 reviews from various Brand Names. They did their experiments on two steps. First, they used balanced data which means that the number of negative reviews (1 and 2 star) is equal to the number of positive reviews (4 and 5 star), and they removed neutral reviews. Second, while using unbalanced data, they considered (1 and 2 stars) as negative reviews and (3, 4, 5) as positive reviews. Our work is directly related to the comments made and the standard is measured by this criterion Support Vector Machine (SVM), BOOSTING, SLDA, NNETWOR, TREE and BAGGING with the aim to come up with the most efficient classified.

According to Tama 2019[28] Final Project built a system that can classify opinions on product reviews into positive and negative sentiments by utilizing the rating. The dataset used is Grocery and Gourmet Food from Amazon as much as 50,000, which will then be labeled using Labeling Methods Average and Binary. The classification of this opinion uses the approach of Supervised Learning Algorithm Multinomial Naïve Bayes. The result of this research shows that labeling using Method Average is suitable for processing Grocery and Gourmet Food Dataset and proves that the best ratio of feature selection usage is 20% succeed to produce 80.48% accuracy. The result of this comparison is good but the method used was limited and could have been done better.

Arif Abdurrahman Farisi[30]. The main objective in this study According to the author It is to evaluate the level of service delivery of hotels is takes a sentiment analysis to quickly detect if the reviews is a positive or negative reviews. The study provides a solution by classifying positive opinion reviews and negative opinions using the Multinomial Naïve Bayes Classifier method and comparing models using preprocessing, feature extraction and feature selection. The best experimental results using preprocessing and feature selection with 10-fold cross validation have an average F1-Score more than 91%. On the data set use is derived from Data Finites' Business Database which contains hotel reviews of as many as 5000 English sentences in csv. The attributes that are in the dataset consist of city, country, hotel name, rating, and review. Then do attribute reduction to fit the research needs. The attribute used is the text review attribute. Then do the manual label according to the review sentence to label 1 (positive) and 0 (negative). The result of manual labeling is 3946 sentences is label 1 and 1053 sentences are label 0. Although this is better than the previous work [28], it is not considered that it is better to gather customers' opinions because the classification is only positive and negative because it does not include neutral comments. What makes our work unique is its ability to provide standards.

Minu P Abraham1 [31]. which is substantially utilized for research purpose the Mobile product review datasets methods are taken from the well-known Kaggle website and it is composed of scripts, accumulation of public datasets, together with a special forum for conversation and collaboration among data scientists working on a given dataset. As the main method the collected dataset of Mobile product reviews is in CSV format. Exhibits some of the features of the explicit aspect extraction corpus. As the main objective it is composed of 200 Amazon reviews of 5 products in the mobile product domain. The fine-grained investigations in identifying the sentiments (opinions) expressed on various aspects of the entity considering the aspects as explicit one over various brands of mobile product reviews and classify these opinions based on some machine learning algorithms. Finally the SVM comparison the result of exploratory outcomes on the different datasets manifests the promising results with respect to the accuracy of classifying the opinions.However, the good results come from the fact that only one machine learning algorithms and its use for different products complicates the task in this regard, it is best that our work be tested with different machine learning algorithms.

Neetu. MS etal [33] .The main objective of this study was to analyze emotional data by analyzing Twitter database data using a variety of machine learning approaches based on a specific domain. They are seen as problems that focus on the problem of recognizing emotional keywords from multiple keywords and mispronouncing spelling and spelling. And their method is based on the characteristic vector based on Naive Bayes, maximum entropy, SVM and stock classifications. Seen as a problem with this work is mainly that it is very different from our work and that they did their work by analyzing it directly in words. But I consider it an input because the method used in this study is similar to our work.

Anurag et al [34]. In this study, they used a unique and unusual methods have introduced a new methodology called combined approach of two separate classifiers called Hidden Markov Models and Support Vector machines. Then the model merges the outcomes of these classifiers using classifier combine rule. This methodology is used to classify the movie reviews relying on the sentiment present in those reviews. The main purpose is to analyze the comments made on the movie. And also they were capable to enhance the anticipated classification results through the use of two classifier association rules. As a result described and presented an approach of handling smileys as well as the slag words, which broadly generate a better sentiment classification with higher accuracy. Although a different approach can be seen in this work, I do not think it is good that the data collected is complex and different website. If our work is clear, the presence of the web and the ability to compare more comments in many ways make it unique.

Sepideh Paknejad [35] The main objective of this study considers the problem of classifying reviews by their overall semantic (positive or negative). The method used to conduct the study two deferent supervised machine learning techniques, SVM and Naïve Bayes, Seen as a problem has been attempted on beauty products from Amazon. Those accuracies have then been compared. The results showed that the SVM approach outperforms the Naiıve Bayes approach when the data set is bigger. However, both algorithms reached promising accuracies of at least 80%. Dataset used the file was converted to the Comma Separated Values (CSV) format, as it is more convenient for python to handle this type of files. However it raises doubts about the accuracy of the model used to compare and contrast the products. The Competition method used is very common. In this regard, our work will be batteries because the product has its own unique users and comments that are directly related to the company.

2.7. Summary of the chapter

The review showed that machine learning, ontology based and lexicon based are the commonly used approaches to deal with sentiment mining. The works reviewed indicated that the approaches except the machine learning rely on tagged list of positive or negative sentiment terms to identify the polarity of terms. On most of reviewed research there are attribute like emoticon expressions, reviewers'. We consider such attributes on this research. In our research we include of the user with text for prediction.

The machine learning technique is based on the concept of training the machine to learn to classify opinionated texts into predefined categories of positive, negative or neutral. Currently, the major practical use of opinion mining for both business and governments is as a means of their customer reviews. They also constitute an arena where the issues of the day are frequently debated and where opinions can be formed on a wide range of topics. Hence many large organizations now have social media teams in their communications or public relations departments which both monitor current events on social networks and actively release content to those networks. A lot of research has been done so far on opinion mining related to improving business by reviewing user opinion commented on product. They develop a methodology to detect product quality and company's' profile using sentiment and lexical analysis. As my knowledge and reviewing a related research using user opinion for predicting in our country

Ethiopia is rare. In this research use the previous work on opinion mining as a guide and different data mining techniques as methodology to develop predictive model, which evaluates and predict user's satisfactions. What makes the work different from the others is that the statistical analysis tool based on R programming language developed we use is largely untested and the result is the best. According to Stat Counter website, 69% of Ethiopian Internet users are Facebook users. Therefore, this survey takes into account the user reality our work is directly related to the comments made and the standard is measured by this criterion Support Vector Machine (SVM), BOOSTING, SLDA, NNETWOR, TREE and BAGGING with the aim to come up with the most efficient classified.

Table 3 Summary of related work

Author, year	Objectives	Methods	Key finding	remark
Alemu Molla, 2015	Find and classify deceptive opinions of tweets that are targeted at a specific product in using user opinions about different Samsung products.	Use the corpus to train the sentiment classifier	detailed study of a recently collected corpus, its basic statistics, and a proposed classification methods for sentiment analysis on twitter messages	
Zerihun Tolla, 2010	Analysis and classification of social media (twitter) user's opinion as past, active, and normal message of sentiment in event.	Machine learning, lexicon-based methods and linguistic analysis.	The analysis of sentiment in the largest spiking events in Twitter posts over a gives strong evidence that important events in Twitter are associated with increases in average negative sentiment strength	
Chen, 2012	Detecting offensive language in social media to protect adolescent online safety	-Appraisal approach -Like n-grams, Bag-of Words.	-Classifying offensive text as strong and weak	
Selama Gebre meskel 2010	Sentiment Mining Model for Opinionated Amharic Texts	Machine learning approach	-Discuss tools, techniques, application of text classification and NLP	
Nizam ani, 2012	Modeling suspicious email detecting using enhanced feature selection	Machine learning algorithm	-Algorithm prediction accuracy improved by feature selection	
Abulai sh, 2017	Activity prediction: Opinion Mining for Customer Review Summarization	Propose two naïve approaches to predicting activities and future Term extraction and matching. And summarization	Predicting popular activities for a later moment based on people's comments feasible,	

Chapter Three

Methodology

3.1. General approach

The aim of predictive classification with the help of machine learning is to develop models that provide efficient prediction result of the target class from the predefined class labels. These analytical models allow analysts and researchers to uncover hidden pattern through learning from historical relationships and trends in the data. In this research project we follow machine learning research methodology. In the field of quantitative research methods, there is a wide variety of statistical and mathematical analysis procedures to choose from. The purpose of opinion mining using the concept of data mining is finding a useful pattern in gathering new level of understanding in connection with algorithms which are used in big data and show the most efficient possible asymptotic consumption of computer resources Opinion mining, is a statistical pattern learning which involves information retrieval to study word frequency distributions, pattern recognition, information extraction, opinion mining techniques including link and association analysis, visualization, and predictive analytics. The complete system consists of the components listed in the following. Each of the main components is explained in more detail.

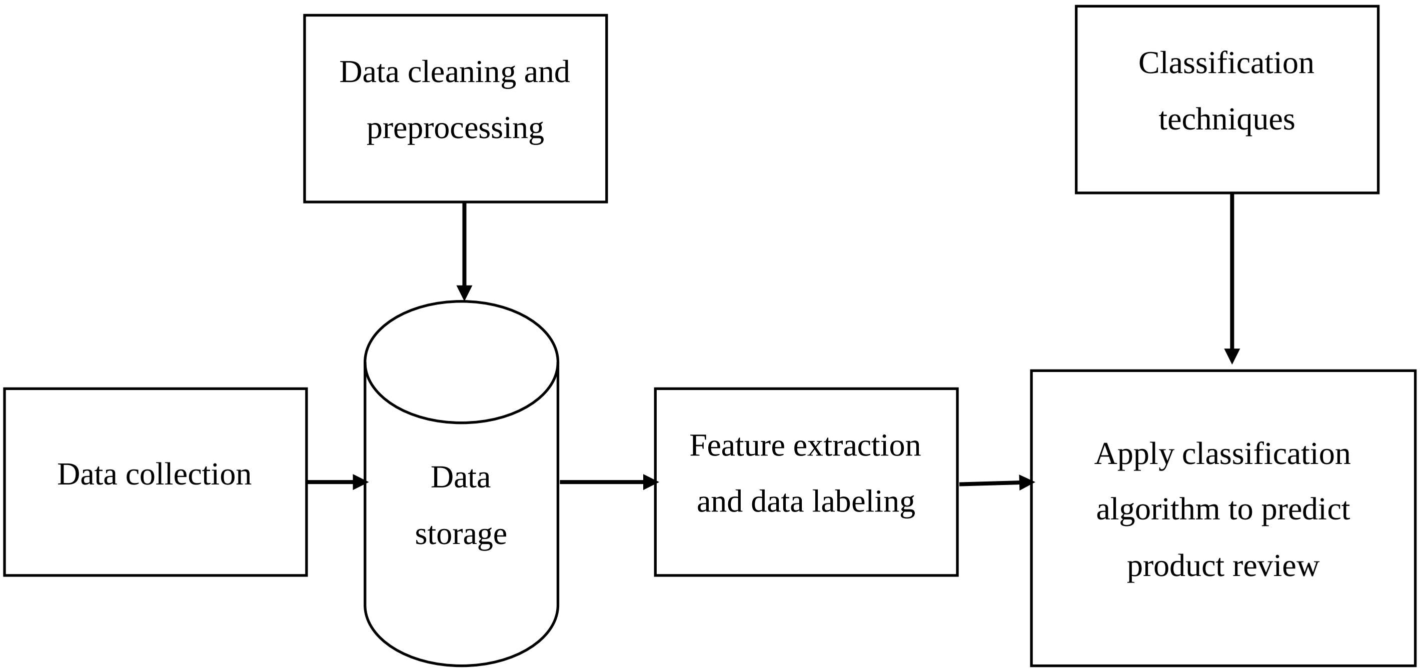

Data cleaning and preprocessing
Classification techniques
Data collection
Data storage
Feature extraction and data labeling
Apply classification algorithm to predict product review

Figure 2 Research process

3.2. Data collection

3.2.1. Data sources

Data source for the research is Facebook. People and companies across disciplines exploit the rich and unique source of data for varied purposes. The major decisive factor for the improvement of the quality of services rendered and enrichment of deliverables are the consumer's opinions. Review sites, blogs and micro blogs provide a good understanding of the reception level of products and services

3.2.2. Review

Sites Opinions are the major and actual data or more precise a decision for any user in making a purchase. The user-generated reviews for products and services are mainly available on internet. The sentiment classification uses reviewer's data are gathered and composed from the websites www.Facebook.com/TECNOMobileEhtiopin, (mobile and product reviews), which hosts of product reviews consumers. Tecno website support from, which is more focused on similar products, which will further support the evaluation process.

3.3. Data cleaning and preprocessing

Data cleaning and preprocessing is done to avoid noisy and inconsistent data. It helps in transforming raw data into an understandable format. Data must be preprocessed in order to

Perform any data mining functionality. Data preprocessing task includes the following tasks: filtering redundant letters from the word, removing questions (WH words), removing special characters, removal of repeated words, removal of non-English words, replacing emoticons, removing of targets mention, removal of hash tags, removal of digits, removal of stop words etc. After irrelevant attribute is removed we load a data to R which we prefer to work on for preprocessing of text such as stop word removing, transformation, vector representation of text etc. Before feeding the dataset to the classifier, an automatic pre-processing procedure assembles the comments for each user and chunks them into sentences.

3.4. Feature extraction and data labeling

Before any classification task using supervised machine learning algorithm, one of the most basic tasks that needs to be done is that of text representation and feature selection. Because the performance of the classification is mainly depend on feature selection. Due to the high dimensionality of text features and the noise features it is important to select best and relevant feature which the machine learning classifier uses to classify user opinion correctly. In general, there are two separate ways of document representation. These are bag of words and strings representation. A bag of words representation is representing a document as a set of words, together with their related frequency in the document. This representation is independent of the sequence of words in the collection. The second approach, strings is to represent text directly as strings, in which each document is a sequence of words. From the above two ways of text representation most text classification task uses the bag-of-words representation as a method to classify because of its simplicity for classification purposes. The most common feature selection which is used in both supervised and unsupervised applications is that of stop-word removal and stemming. In stop-word removal, we determine the common words in the documents which are not useful to classify text to different classes. In stemming, different forms of the same word are merged into a single word. For example, singular, plural and different tenses are consolidated into a single word. However, stemming is not specific and relevant for our classification problem, because we follow a supervised classification task. In case of supervised classification task training data set must be labeled first. We label the datasets to a predefined class is depend on each words and using stemming can change the semantic of our text. In the case of the classification problem, which we are going to use it makes sense to supervise the feature selection process with the use of the class labels. For our purpose as supervised Classification uses a labeled class for training data sets labeling is done manually. Labeling datasets manually takes much time but it must be labeled with carefulness because it has effect on the labeling of test data.

3.5. Classification techniques

In the field of opinion mining several varieties of techniques have been designed for text classification. These techniques of classification generally exist for quantitative or categorical data. Since text may be describe or modeled as quantitative data with frequencies on the word attributes, it is possible to use most of the methods for quantitative data directly for this research project. However, text is a particular kind of data in which the word attributes are sparse, and high dimensional, with low frequencies on most of the words, it is critical to select classification algorithm which effectively account for these characteristics of text. For this research project depending upon our dataset and the performance of their classification we select six machine learning algorithm from the following explained algorithm.

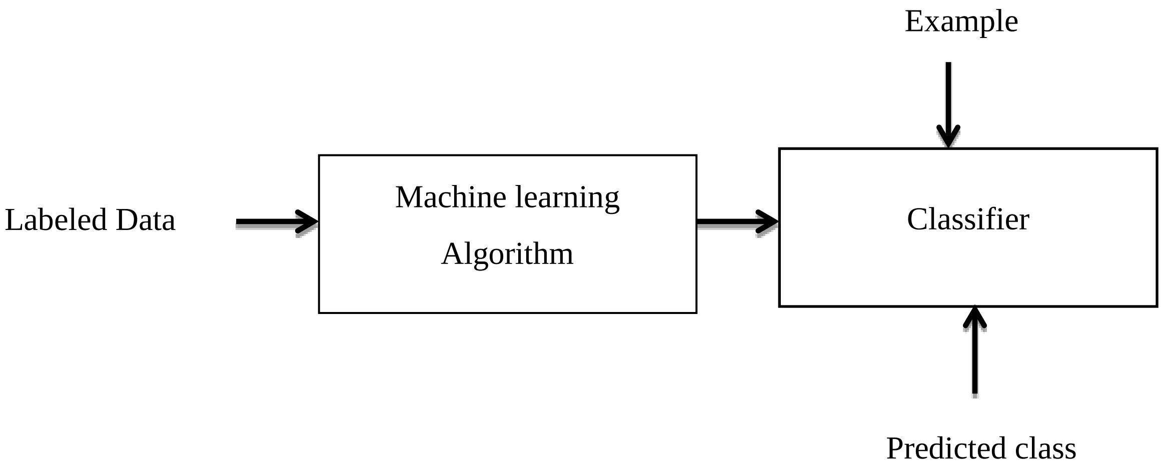


Figure 3 Model building process

Many text classifiers have been proposed in the literature reviewed such as probabilistic models, machine learning techniques, etc. mainly all classifier differ in the approach they follow: decision trees, naıve-Bayes, rule induction, neural networks, nearest neighbors, and lately, support vector machines. Although many approaches have been proposed, automated text classification is still a major area of research primarily because the effectiveness of current automated text classifiers is not faultless and still needs improvement. Based on the aim of our classification task and the type our data set we select to use the following machine learning classifier.

3.5.1. SVM Classifiers

The first text classifier used for our classification task is SVM (support vector machine) Classifiers. These classifiers are attempting to divide the data space with the use of linear or nonlinear demarcations between the different classes. The key in such SVM classifiers is to determine the optimal boundaries between the different classes and use them for the aim of classification. The application of SVM approach to text classification has been propose by [36]. The support vector machine classifier needs both positive and negative training set which are mostly not common for other classification methods. These positive and negative training set are needed for this classifier to seek for the decision surface that best separates the positive from the negative data in the n dimensional space, so called the hyper plane. The document representatives which are closest to the decision surface are called the support vector.

3.5.2. Neural Network Classifiers

Neural networks classifier is used in a wide area of text classification purpose. In the context of opinion text the main difference for these classifiers is to of domains for the purposes of classification. In the context of text data, the main difference for neural network classifiers is to familiarize these classifiers with the use of word features. Over all neural network classifiers are related to support vector machine classifiers; indeed, they both are in the category of discriminative classifiers, which are in contrast with the generative classifiers [29]. A neural network classifier is a network of units, where the input units usually represent terms and the output unit represents the category. For classifying a test document, its term weights are assigned to the input units the activation of these units is propagated forward through the network, and the value that the output unit takes up as a consequence determines the categorization decision. Some of the researches use the single-layer perceptron, due to its simplicity of implementing [37]. For the neural network classifier to classify with efficient performance we must use efficient feature selection method which reduce the dimension as well [38].

3.5.3. Bagging

Bagging, a Parallel ensemble method (stands for Bootstrap Aggregating), is a way to decrease the variance of the prediction model by generating additional data in the training stage. This is produced by random sampling with replacement from the original set. [39] By sampling with replacement, some observations may be repeated in each new training data set. In the case of Bagging, every element has the same probability to appear in a new dataset. By increasing the size of the training set, the model's predictive force can't be improved. It decreases the variance and narrowly tunes the prediction to an expected outcome. These multisite of data are used to train multiple models. As a result, we end up with an ensemble of different models. The average of all the predictions from different models is used. This is more robust than a model. Prediction can be the average of all the predictions given by the different models in case of regression. In the case of classification, the majority vote is taken into consideration.

3.5.4. Boosting

Builds Boosting is a sequential ensemble method that in general decreases the bias error and builds strong predictive models.[39][40] The term 'Boosting' refers to a family of algorithms which converts weak comments to a strong comments boosting gets multiple comments. The data samples are weighted and therefore, some of them may take part in the new sets more often. In each iteration, data points that are miss predicted are identified and their weights are increased so that the next comment pays extra attention to get them right.

3.5.5. Supervised latent Dirichlet allocation (sLDA)

sLatent Dirichlet allocation (LDA) [41] is an unsupervised latent variable model originally applied in the field of document modeling due to its ability to decompose documents into topics and uncover topics decomposition into words in a concise manner. As an unsupervised model, LDA can be used to perform dimensionality reduction by mapping the high dimensional bag-of-words representation to lower dimensional topic representation.

3.5.6. Tree

Decision tree is a type of supervised learning algorithm (having a predefined target variable) that is mostly used in classification problems [42]. It works for both categorical and continuous input and output variables. In this technique, we split the comments or sample into two or more sets (or sub-comments) based on most significant splitter / differentiator in input variables.

3.6. Evaluation of the model

The choice of evaluation method for machine learning classifier we used is a critical step. The classifier's evaluation is most often based on prediction accuracy that is the percentage of correct prediction divided by the total number of predictions. For this research project we use two general classifier evaluation techniques. The first technique we use to evaluate accuracy of the classifier is to split the training datasets in to two using a rule two thirds of the data sets for training and one third to estimate the performance. The second techniques we use are cross-validation. Because of the size of our dataset we use tenfold cross validation. In case of cross validation technique, the training set is divided into mutually exclusive and equal-sized subsets and for each subset the classifier is trained on the union of all the other subsets. The average of the error rate of each subset is therefore an estimate of the error rate of the classifier. A variety of factors must be examined if performance of the classifier is not satisfactory. One cause of performance degradation may be relevant features for the problem are not being used, the dimensionality of the problem is too high, a larger training set is needed and the selected algorithm is inappropriate or parameter tuning is needed. Another problem could be that the dataset is imbalanced classes for training. Performance evaluation during the task of text classification is typically conducted experimentally, rather than analytically. The experimental evaluation of classifiers, rather than concentrating on issues of Efficiency, usually tries to evaluate the effectiveness of a classifier, i.e. its capability of taking the right categorization decisions. The research done so far on machine learning classifier propose an approach like Precision and recall [32]; fallout, error, accuracy etc. are stated below

- Precision (P) is defined as the number of true positives ($T_{P)}$ over the number of true positives plus the number of false positives (F_P).
- Recall (R) is defined as the number of true positives (T_P) over the number of true positives plus the number of false negatives (F_P).

These quantities are also related to the (F_1) score, which is defined as the harmonic mean of precision and recall.

Accuracy = TP + TN

3.7. Tools

A number of open source and proprietary based data mining and data visualization tools exist which are used for information extraction from large data repositories and for data analysis. Some of the data mining tools which exist in the market are Weka, Rapid MinerOrange, R, KNIME, ELKI, GNU Octave, Apache Mahout, SCaViS, Natural Language Toolkit, Tableau, etc.

3.7.1. The Rapid Miner Tool

Rapid Miner is one of the most widely used open source data mining tool developed in 2001 by Ingo Mierswa and Ralf Klinkenberg. Prior to 2006, it was known as YALE (Yet another Learning Tool) [43]. Rapid Miner is a XML based data mining tool used to implement various machine learning and data mining processes. It is a popular tool to implement classification and clustering algorithms. An important feature of Rapid Miner is its ability to display results visually. “Rapid Miner provides learning schemes and models and algorithms from Weka and R scripts that can be used through extensions.”

3.7.2. The Weka Tool

Weka is one of the very popular open source data mining tools developed at the University of Waikato in New Zealand in 1992. It is a Java based tool and can be used to implement various machine learning and data mining algorithms written in Java [44]. The simplicity of using Weka has made it a landmark for machine learning and data mining implementation [45]. Weka supports reading of files from several different databases and also allows importing the data from the internet, from web pages or from a remotely located SQL database server by entering the URL of resource. Among all the available data mining tools, Weka is the most commonly used of all due to its fast performance and support for major classification and clustering algorithm. Weka can be easily downloaded and deployed. Weka performs accurately when the size of the data set is not large. If the size is large, then Weka does experience some performance issues.

3.7.3. The R Programming Tool

R is an open source statistical analysis tool based on C and FORTRAN programming language developed by Ross Ihaka and Robert Gentleman at the University Of Auckland, New Zealand [44]. R was released in 1997 and it is currently licensed using GNU General Public License. Using R, well-designed publication-quality plots can be produced, including mathematical symbols and formulae wherever required. R uses a number of different packages to support data mining or statistics and provides a well-integrated collection of intermediate tools for data analysis. R-Integration can be used in combination with Tableau to create visual representations of various data mining algorithms due to the clear and interactive visualizations which can be created using Tableau. Although R provides less support to data mining algorithms as compared to Rapid Miner and Weka, it does implement a few data mining algorithms. R supports implementation of Naïve Bayes algorithm, confusion matrix and summary of data which is used for classification of data. R uses a code-driven methodology which involves use of a number of in-built functions and commands to perform statistical analysis and data mining.

For this research instead of using other tools we prefer to use R programming tools. R is out of the most efficient tool used for statistical computation and graphics. It has a capability of using a programming language, high level graphics, interfaces to other languages and debugging facilities.

Chapter Four

Preparing dataset and Preprocessing

4.1. Overview

The proposed general framework for the research project is shown on the following figure. it covers the different stages in data preprocessing and preparation. The presented general framework fits a broad variety of datasets. Raw data prior to cleaning and preparation is usually not ready for distilling correct inferences. Each of the main components is explained in more detail.

4.2. Data description

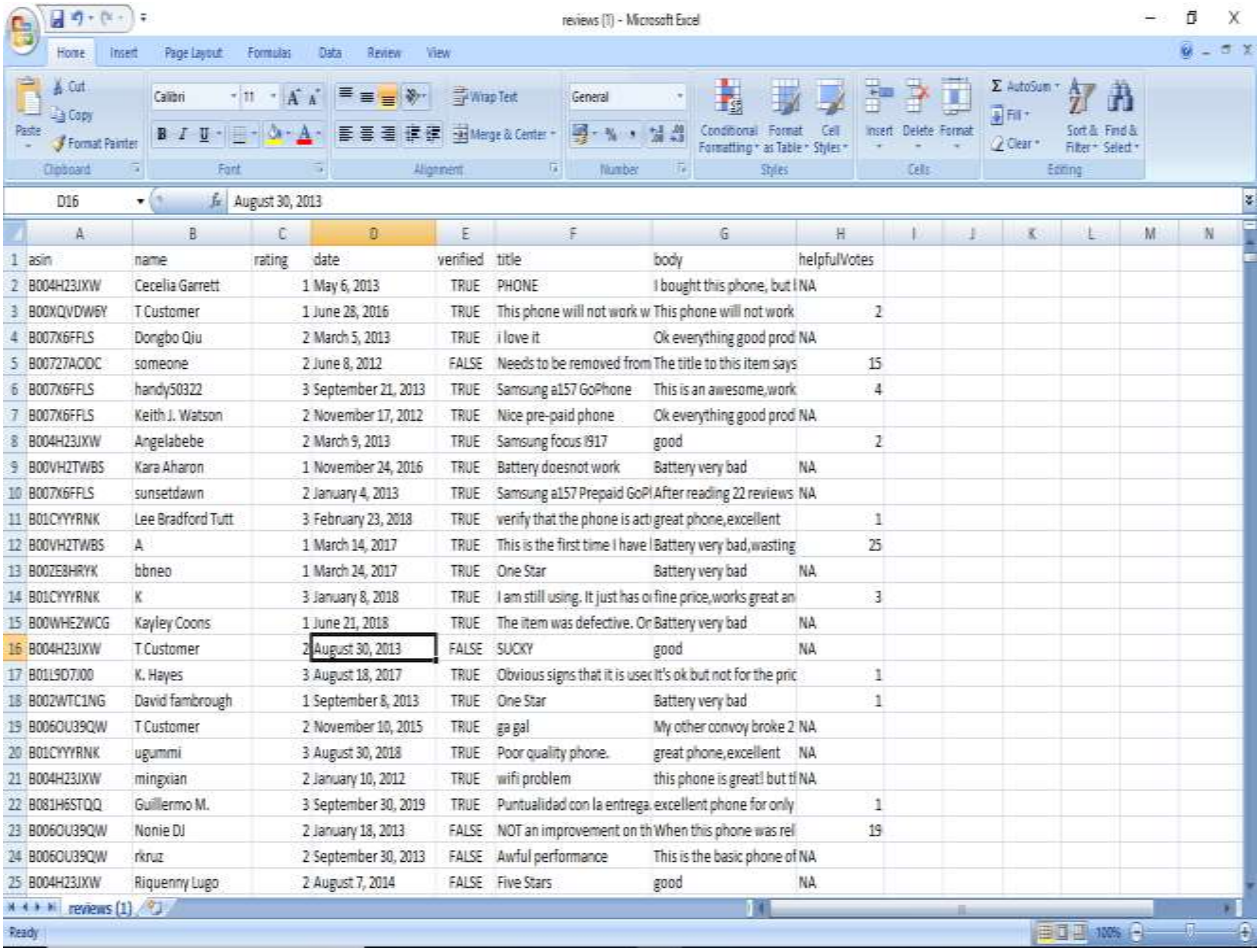
The purpose of opinion mining using the concept of data mining is finding a useful pattern in gathering new level of understanding in connection with algorithms which are used in big data and show the most efficient possible asymptotic consumption of computer resources. Opinion mining is a statistical pattern learning which involves information retrieval to study word frequency distributions, pattern recognition, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. For This paper, the data is collected from web. The data is a customer review Meta data of different types of cellphone product.

Table 4 Feature descriptions

Feature	Description
Product Name	Product ASIN text_format name sort
Reviewer	Reviewer Name grid_3x3 rating sort
Rating	Reviewer Rating (scale 1 to 5) calendar_today date sort
Review Date	Review Date check verified sort
Reviews	Valid Customer text_format title sort
Title	Review Title text_format body sort

4.3. Data collection

First, we got several datasets from the UCI depositories, Facebook for DevelopersWebsites to extract with R code. The dataset contains reviews about the Cell phone Products of the Websites. The dataset contains about 50,000 to 90,000 of reviews but in our project we have taken only 2000 of reviews and made analysis on it. We can take more number of reviews for analysis but if we do that since we use machine learning on textual data, it takes much time. We extract a Dataset in JSON format later we converted the files into one extension i.e... CSV file. Csv file is Comma Separated File where data is separated by comma and saved in dot csv.The data we got is presented in irregular way so to convert the date in the structured manner we need Pre-processing technique.



D16 | August 30, 2013

	A	B	C	D	E	F	G	H
1	asin	name	rating	date	verified	title	body	helpfulVotes
2	B004H23JXW	Cecelia Garrett	1	May 6, 2013	TRUE	PHONE	I bought this phone, but I	NA
3	B00XQVDW6Y	T Customer	1	June 28, 2016	TRUE	This phone will not work w	This phone will not work	2
4	B007X6FFLS	Dongbo Qiu	2	March 5, 2013	TRUE	i love it	Ok everything good prod	NA
5	B00727AODC	someone	2	June 8, 2012	FALSE	Needs to be removed from	The title to this item says	15
6	B007X6FFLS	handy50322	3	September 21, 2013	TRUE	Samsung a157 GoPhone	This is an awesome,work	4
7	B007X6FFLS	Keith J. Watson	2	November 17, 2012	TRUE	Nice pre-paid phone	Ok everything good prod	NA
8	B004H23JXW	Angelabebe	2	March 9, 2013	TRUE	Samsung focus I917	good	2
9	B00VH2TWBS	Kara Aharon	1	November 24, 2016	TRUE	Battery doesnot work	Battery very bad	NA
10	B007X6FFLS	sunsetdawn	2	January 4, 2013	TRUE	Samsung a157 Prepaid GoPl	After reading 22 reviews	NA
11	B01CYYYRNK	Lee Bradford Tutt	3	February 23, 2018	TRUE	verify that the phone is act	great phone,excellent	1
12	B00VH2TWBS	A	1	March 14, 2017	TRUE	This is the first time I have I	Battery very bad,wasting	25
13	B00ZE8HRYK	bbneo	1	March 24, 2017	TRUE	One Star	Battery very bad	NA
14	B01CYYYRNK	K	3	January 8, 2018	TRUE	I am still using. It just has o	fine price,works great an	3
15	B00WHE2WCG	Kayley Coons	1	June 21, 2018	TRUE	The item was defective. Or	Battery very bad	NA
16	B004H23JXW	T Customer	2	August 30, 2013	FALSE	SUCKY	good	NA
17	B01L9D7J00	K. Hayes	3	August 18, 2017	TRUE	Obvious signs that it is usec	It's ok but not for the pric	1
18	B002WTC1NG	David fambrough	1	September 8, 2013	TRUE	One Star	Battery very bad	1
19	B006OU39QW	T Customer	2	November 10, 2015	TRUE	ga gal	My other convoy broke 2	NA
20	B01CYYYRNK	ugummi	3	August 30, 2018	TRUE	Poor quality phone.	great phone,excellent	NA
21	B004H23JXW	mingxian	2	January 10, 2012	TRUE	wifi problem	this phone is great! but tl	NA
22	B081H6STQQ	Guillermo M.	3	September 30, 2019	TRUE	Puntualidad con la entrega.	excellent phone for only	1
23	B006OU39QW	Nonie DJ	2	January 18, 2013	FALSE	NOT an improvement on th	When this phone was rel	19
24	B006OU39QW	rkruz	2	September 30, 2013	FALSE	Awful performance	This is the basic phone of	NA
25	B004H23JXW	Riquenny Lugo	2	August 7, 2014	FALSE	Five Stars	good	NA

Figure 4 Screen shot of Collected data

4.4. Pre-processing

Preprocessing is done to avoid noisy and inconsistent data. It helps in transforming raw data into an understandable format. Data must be preprocessed in order to perform any data mining functionality. Data description each user opinion is extracted with different entity. The first step of data preprocessing is to clean this data which is important for classification. Remove all entities which are necessary for our experiment. After removing all entities, the documents should be transformed into a representation suitable for applying the learning algorithms. In addition to removing irrelevant entities perform the following Preprocessing and comment representation phase, which is implemented using R package.

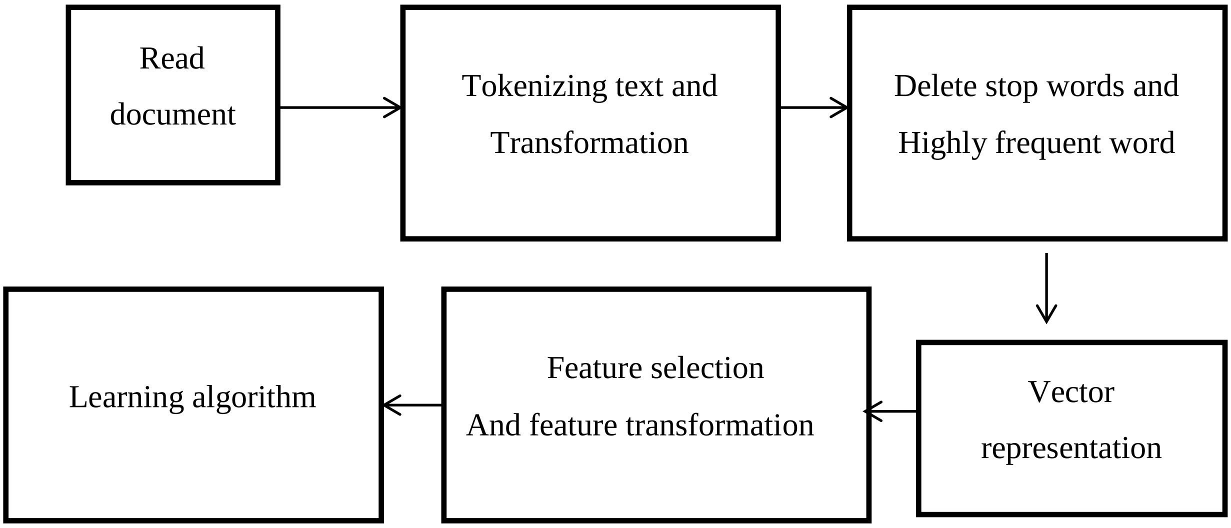
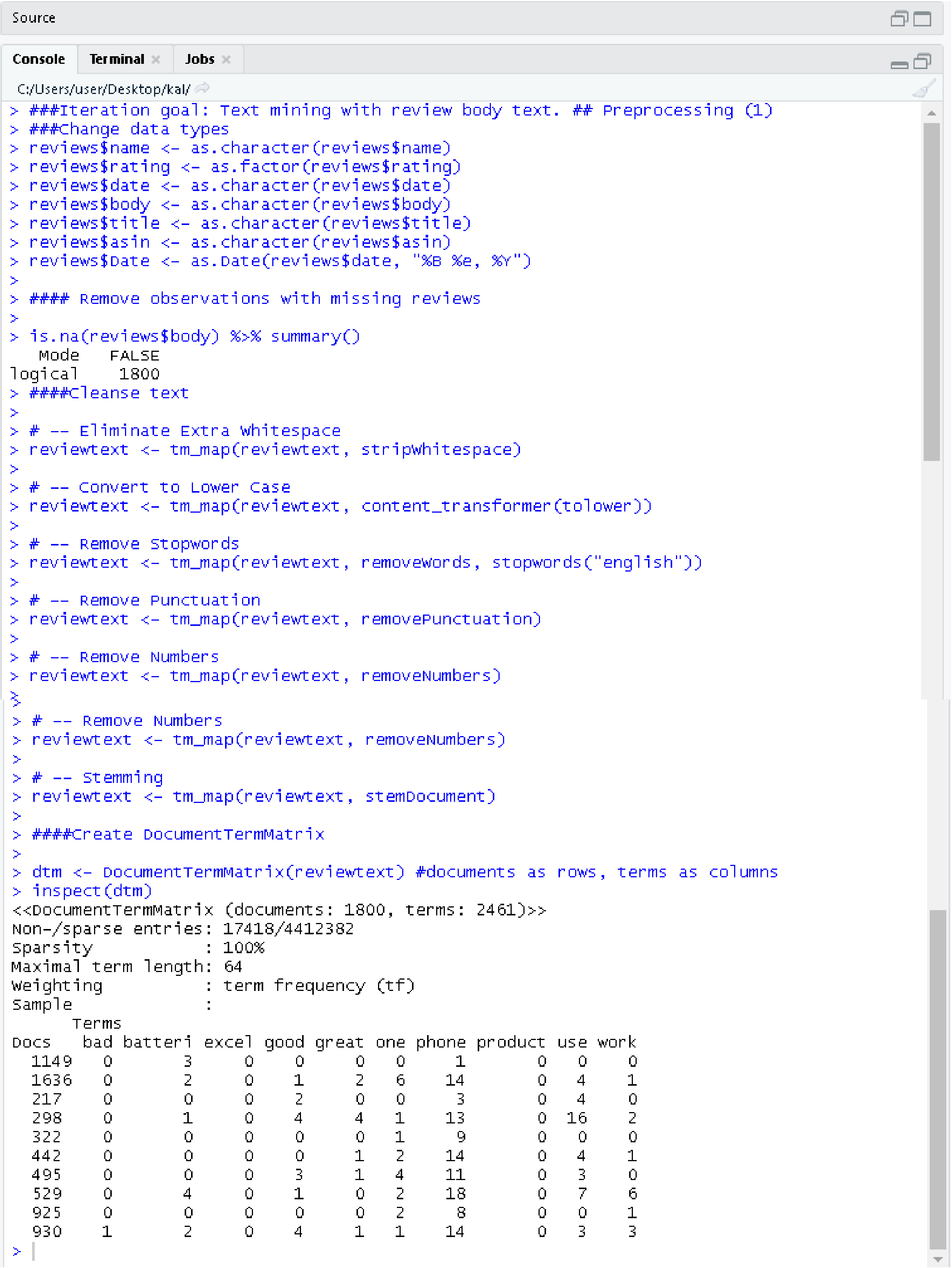

Read document
Tokenizing text and Transformation
Delete stop words and Highly frequent word
Vector representation
Feature selection And feature transformation
Learning algorithm

Figure 5 preprocessing steps



```
> ###Iteration goal: Text mining with review body text. ## Preprocessing (1)
> ###Change data types
> reviews$name <- as.character(reviews$name)
> reviews$rating <- as.factor(reviews$rating)
> reviews$date <- as.character(reviews$date)
> reviews$body <- as.character(reviews$body)
> reviews$title <- as.character(reviews$title)
> reviews$asin <- as.character(reviews$asin)
> reviews$Date <- as.Date(reviews$date, "%B %e, %Y")
>
> #### Remove observations with missing reviews
>
> is.na(reviews$body) %>% summary()
   Mode   FALSE
logical    1800
> ####Cleanse text
>
> # -- Eliminate Extra Whitespace
> reviewtext <- tm_map(reviewtext, stripWhitespace)
>
> # -- Convert to Lower Case
> reviewtext <- tm_map(reviewtext, content_transformer(tolower))
>
> # -- Remove Stopwords
> reviewtext <- tm_map(reviewtext, removeWords, stopwords("english"))
>
> # -- Remove Punctuation
> reviewtext <- tm_map(reviewtext, removePunctuation)
>
> # -- Remove Numbers
> reviewtext <- tm_map(reviewtext, removeNumbers)
>
> # -- Remove Numbers
> reviewtext <- tm_map(reviewtext, removeNumbers)
>
> # -- Stemming
> reviewtext <- tm_map(reviewtext, stemDocument)
>
> ####Create DocumentTermMatrix
>
> dtm <- DocumentTermMatrix(reviewtext) #documents as rows, terms as columns
> inspect(dtm)
<<DocumentTermMatrix (documents: 1800, terms: 2461)>>
Non-/sparse entries: 17418/4412382
Sparsity           : 100%
Maximal term length: 64
Weighting          : term frequency (tf)
Sample             :
      Terms
Docs   bad batteri excel good great one phone product use work
  1149   0       3     0    0     0   0     1       0   0    0
  1636   0       2     0    1     2   6    14       0   4    1
  217    0       0     0    2     0   0     3       0   4    0
  298    0       1     0    4     4   1    13       0  16    2
  322    0       0     0    0     0   1     9       0   0    0
  442    0       0     0    0     1   2    14       0   4    1
  495    0       0     0    3     1   4    11       0   3    0
  529    0       4     0    1     0   2    18       0   7    6
  925    0       0     0    0     0   2     8       0   0    1
  930    1       2     0    4     1   1    14       0   3    3
>
```

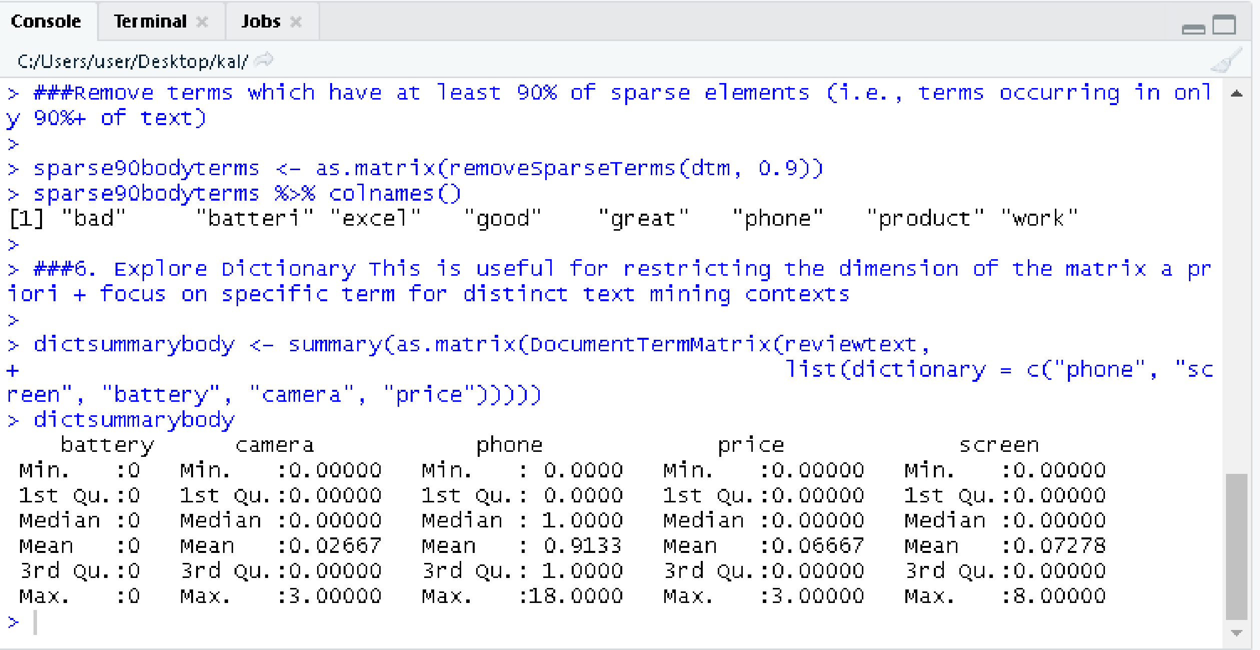
Figure 6 screenshot of preprocessing

4.4.1. Transformation and tokenizing text

In this step, all the HTML or SGML mark-up tags and non-alpha characters are removed from the comments in the data. After removing every irrelevant attribute which is not useful for our classification, the next step is converting all the characters in a comments (message) into the same case, that is converting all the characters into lower-case so that Tokens consisting of alpha characters are extracted.

4.4.2. Removing Stop words and highly frequent words

There are words in English which is not useful for the classification task. These words are: pronouns, prepositions and conjunctions that are used to provide structure in the language rather than content. These words, which are come across very frequently, carry no useful information about the content and thus the category of documents are called stop words. Removing stop words from the documents is very common in information retrieval. We have decided to eliminate the stop words from the documents, which lead to a drastic reduction in the dimensionality of the feature some space. To remove stop words we use table look up. I.e. by referring a table words which belong to the lookup table is removed.

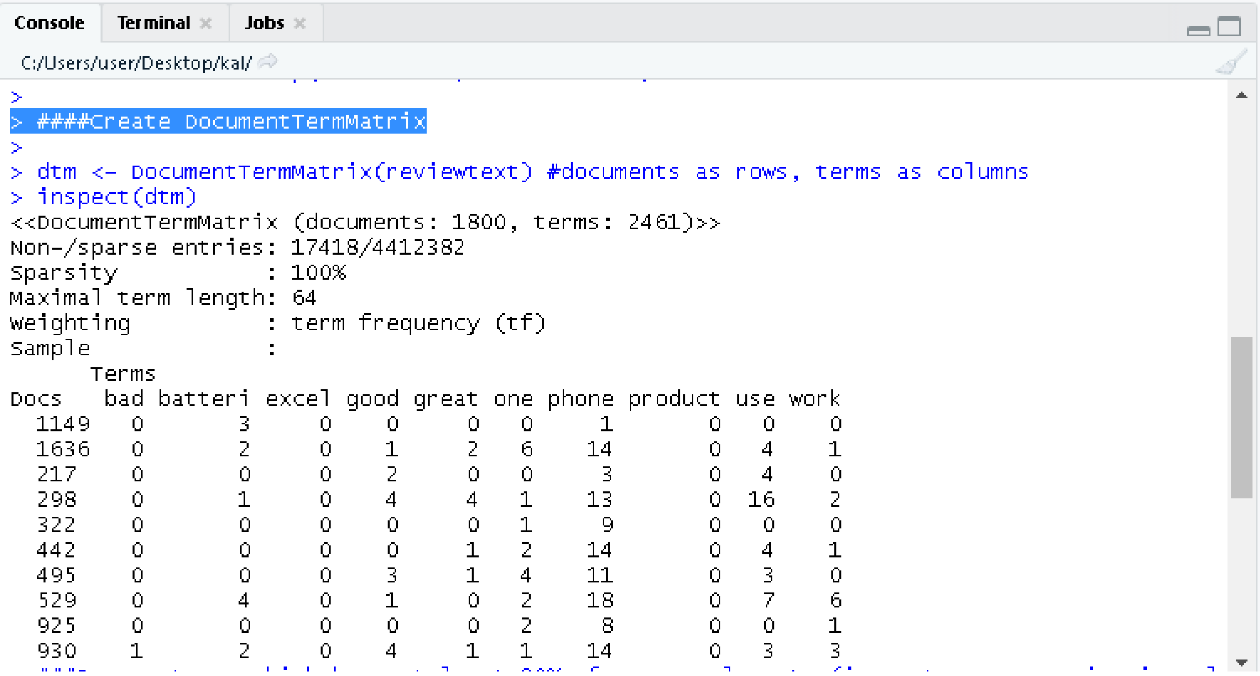


```
> ###Remove terms which have at least 90% of sparse elements (i.e., terms occurring in onl
y 90%+ of text)
>
> sparse90bodyterms <- as.matrix(removeSparseTerms(dtm, 0.9))
> sparse90bodyterms %>% colnames()
[1] "bad"      "batteri" "excel"    "good"     "great"    "phone"    "product" "work"
>
> ###6. Explore Dictionary This is useful for restricting the dimension of the matrix a pr
iori + focus on specific term for distinct text mining contexts
>
> dictsummarybody <- summary(as.matrix(DocumentTermMatrix(reviewtext,
+                                                          list(dictionary = c("phone", "sc
reen", "battery", "camera", "price")))))
> dictsummarybody
    battery        camera              phone               price               screen
 Min.   :0    Min.   :0.00000   Min.   : 0.0000   Min.   :0.00000   Min.   :0.00000
 1st Qu.:0    1st Qu.:0.00000   1st Qu.: 0.0000   1st Qu.:0.00000   1st Qu.:0.00000
 Median :0    Median :0.00000   Median : 1.0000   Median :0.00000   Median :0.00000
 Mean   :0    Mean   :0.02667   Mean   : 0.9133   Mean   :0.06667   Mean   :0.07278
 3rd Qu.:0    3rd Qu.:0.00000   3rd Qu.: 1.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
 Max.   :0    Max.   :3.00000   Max.   :18.0000   Max.   :3.00000   Max.   :8.00000
> 
```

Figure 7 screenshot of removing terms

4.4.3. Document Indexing and vector representation

The main objective of document indexing during the classification task is to increase the efficiency classification by extracting from the resulting document a selected set of terms to be used for indexing the document. The task of Document indexing is choosing the appropriate set of keywords based on the whole corpus of documents and assigning weights to those selected keywords for each particular document so that each document is transformed in to a vector of keyword weights. The weight is related to the frequency of occurrence of the term in the document and the number of documents that use that term. Generally, documents representation is one way reducing the complexity of the documents which make them easier to handle. That is the document have to be transformed the original text document to a document vector. Perhaps most commonly used document representation is called vector space model. Vector space model, documents are represented by vectors of words. Usually, one has a collection of documents which is represented word by word document Matrix as shown in the following.



```
>
> ####Create DocumentTermMatrix
>
> dtm <- DocumentTermMatrix(reviewtext) #documents as rows, terms as columns
> inspect(dtm)
<<DocumentTermMatrix (documents: 1800, terms: 2461)>>
Non-/sparse entries: 17418/4412382
Sparsity           : 100%
Maximal term length: 64
Weighting          : term frequency (tf)
Sample             :
      Terms
Docs   bad batteri excel good great one phone product use work
  1149   0       3     0    0     0   0     1       0   0    0
  1636   0       2     0    1     2   6    14       0   4    1
  217    0       0     0    2     0   0     3       0   4    0
  298    0       1     0    4     4   1    13       0  16    2
  322    0       0     0    0     0   1     9       0   0    0
  442    0       0     0    0     1   2    14       0   4    1
  495    0       0     0    3     1   4    11       0   3    0
  529    0       4     0    1     0   2    18       0   7    6
  925    0       0     0    0     0   2     8       0   0    1
  930    1       2     0    4     1   1    14       0   3    3
```

Figure 8 screenshot of document matrix

4.5. Feature selection

The main objective feature selection method during the classification task of text document is to reduce the dimension of the datasets by removing feature which is irrelevant for classification purpose. This reduction of irrelevant feature has a number of advantage such as decreasing the size of datasets and reduce computational time for the text classifier algorithm that do not scale well the feature set size. In doing dimensionality reduction we can improve accuracy of classification. Another important benefit of feature selection is its capability to reduce over fitting, i.e. the occurrence by which a classifier is adjusted to the contingent characteristics of the training datasets rather than the constitutive characteristics of the categories, and therefore, to increase generalization. As previous work on document classification task proposed the most widely used method for document representation is the vector space model, which we have also decided to use in our feature selection task. In this case each document is represented as a vector d. Each dimension in the vector d stands for a distinct term in the term space of the document collection. A term in the document collection can stand for a distinct single-word or a phrase. Many evaluation metrics for feature have been proposed so far in the literature. In vector space representation, defining terms as distinct single words is referred to as “bag of words” representation which we use in this research project. Using document as “bag of words” representation is the most frequently used method for defining terms and it is computationally more efficient than the phrase representation, we have chosen to adapt this method to define terms of the feature space.

Chapter Five

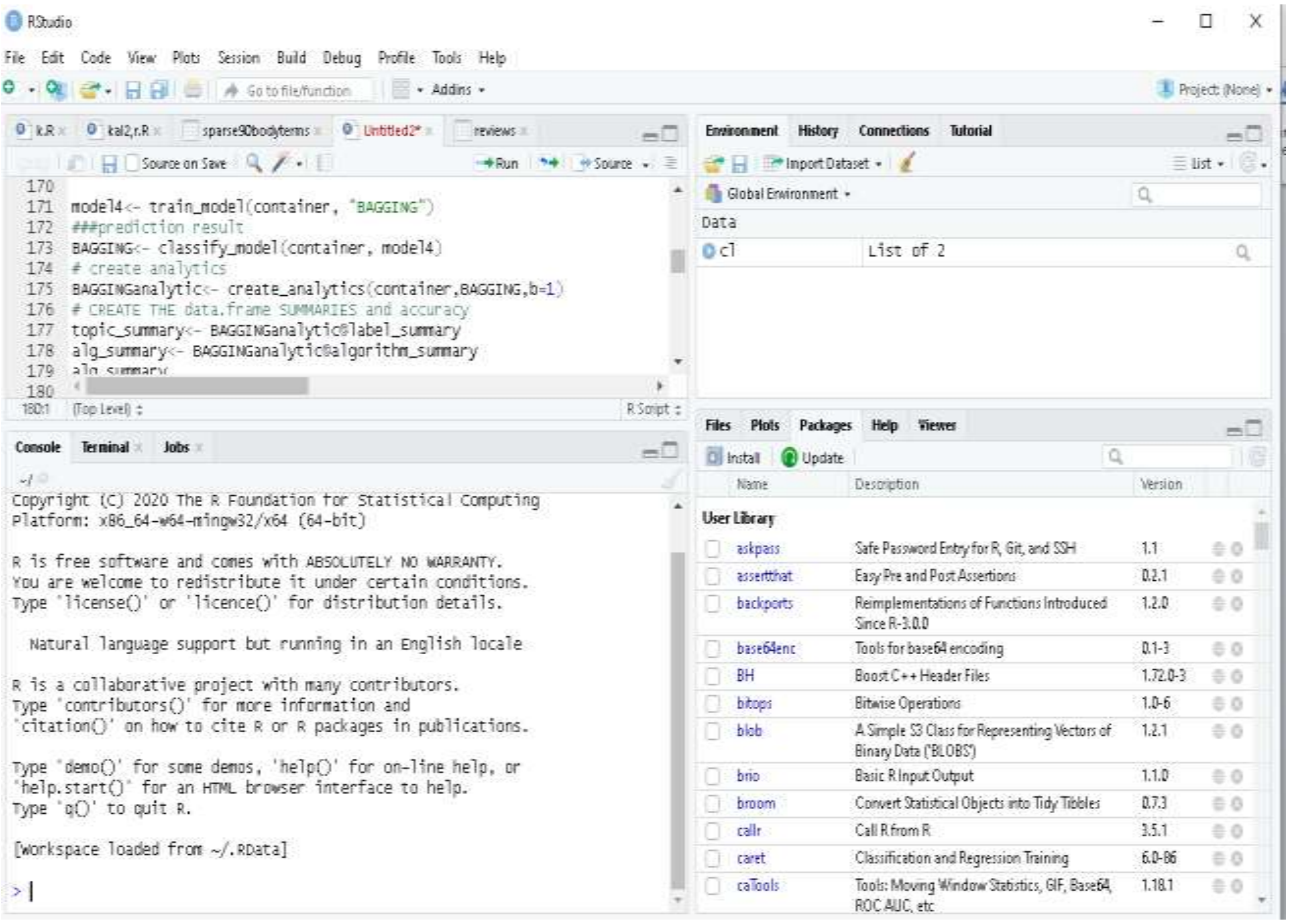
Experimentations and Evaluation of Results

5.1. Overview

We experimentally tested our approach using datasets that we collected from Tecno Facebook page. The datasets were collected using Facebook Graph API Explorer & R. The implementation of our system requires certain R packages. We have imported the necessary packages and load required library. We used R Studio as a platform for implementation. In order to see whether the extracted customer reviews can be predicted or not we prepare a data set containing 2000 rows and 8 columns.

5.2. Experimental Setting

As the first step in developing predictive classification model, select the actual modeling technique that is to be used. As stated in chapter 3 and chapter 4 we prepare a datasets of user opinion collected from Facebook page and run this datasets on Rstudio development environment withRTextTools package and TM package using object oriented programming code. RTextTools is a machine learning package for automatic text classification that makes it simple for users having a little knowledge of objects oriented programming language to get started with machine learning, while allowing experienced users to easily experiment with different settings and algorithm combinations. We build different model by varying attributes and evaluate each model. All experiments are first run on the dataset using 10-fold cross validation using cross_validate() function in RTextTools library to test the validity of our data sets on each algorithm



```
model4<- train_model(container, "BAGGING")
###prediction result
BAGGING<- classify_model(container, model4)
# create analytics
BAGGINGanalytic<- create_analytics(container,BAGGING,b=1)
# CREATE THE data.frame SUMMARIES and accuracy
topic_summary<- BAGGINGanalytic@label_summary
alg_summary<- BAGGINGanalytic@algorithm_summary
alg_summary
```

```
~/
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> |
```

Data

cl	List of 2

User Library

	Name	Description	Version
□	askpass	Safe Password Entry for R, Git, and SSH	1.1
□	assertthat	Easy Pre and Post Assertions	0.2.1
□	backports	Reimplementations of Functions Introduced Since R-3.0.0	1.2.0
□	base64enc	Tools for base64 encoding	0.1-3
□	BH	Boost C++ Header Files	1.72.0-3
□	bitops	Bitwise Operations	1.0-6
□	blob	A Simple S3 Class for Representing Vectors of Binary Data ('BLOBS')	1.2.1
□	brio	Basic R Input Output	1.1.0
□	broom	Convert Statistical Objects into Tidy Tibbles	0.7.3
□	callr	Call R from R	3.5.1
□	caret	Classification and Regression Training	6.0-86
□	caTools	Tools: Moving Window Statistics, GIF, Base64, ROC AUC, etc	1.18.1

Figure 9 Rstudio working environment

5.3. Experimentation and Model Building

Based on techniques mentioned on section 5.1 and 5.2 series of experiments were conducted in order to get the best output and best algorithm that classifies user feedback collected from Tecno facebook page. Goal of the experiments are Predicting the star rating ("rating") given by a user based on their review ("body").

The experimentation is divided in to two phases the first phase includes all experimentation with 10 fold cross validation and the second phase includes 75/25 percentage split by selected algorithms from phase one. We calculated the mean accuracy across all folds for estimating generalization performance. For evaluation purpose we use slot analytics stored in RTextTools package. Following are summary of slot stored in RTextTools which we use to evaluate performance of the model.

- ✓ label_summary Object of class "data.frame": stores the analytics for each label, including the percent coded accurately and how much over coding occurred
- ✓ document_summary Object of class "data.frame": stores the analytics for each document, including all available raw data associated with the learning process

- ✓ algorithm_summary Object of class "data.frame": stores precision, recall, and F-score statistics for each algorithm, broken down by label

- ✓ ensemble_summary Object of class "matrix": stores the accuracy and coverage for an nalgorithmensemblescorining

Phase one

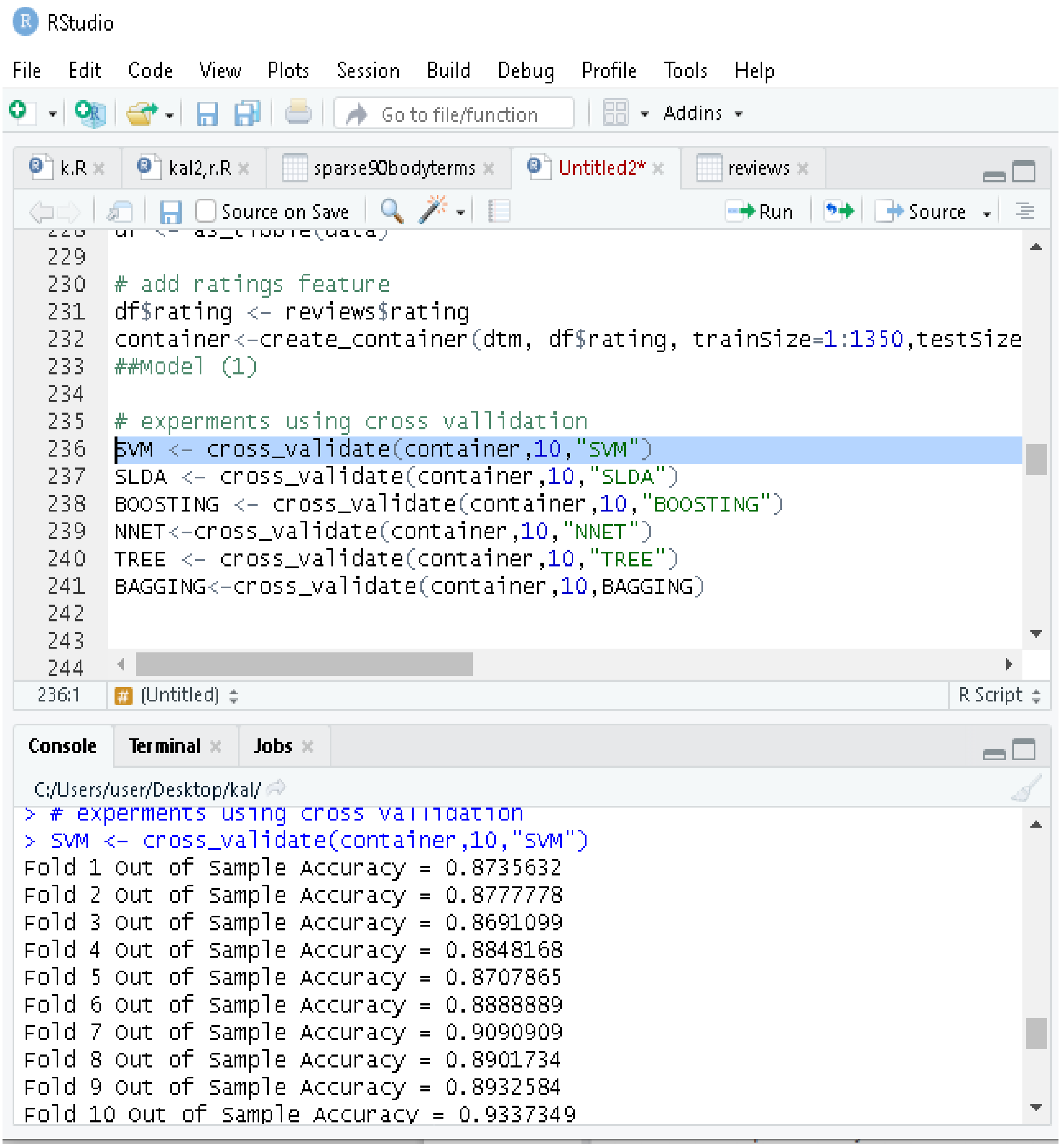
In this phase all experimentations using algorithms mentioned in section 5.2 are conducted using 10 fold cross validation and the model with best performance based on evaluation metrics can be subjected to phase two which is 75/25 percentage split.

All the experiments done follow the following steps in common

- Setting up the project
- Explore data
- Start
  - Text mining (1): body text
  - Prepare data for modeling (1)
  - Model

Experiment 1

This experiment is done using SVM classifier with 1800 datasets using 10-fold cross validation. 10 fold Cross-validation break data into 10 sets of size n/10, Train on 9 datasets and test on 1. Repeat 10 times and take a mean accuracy.



```
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##Model (1)

# experments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
BAGGING<-cross_validate(container,10,BAGGING)
```

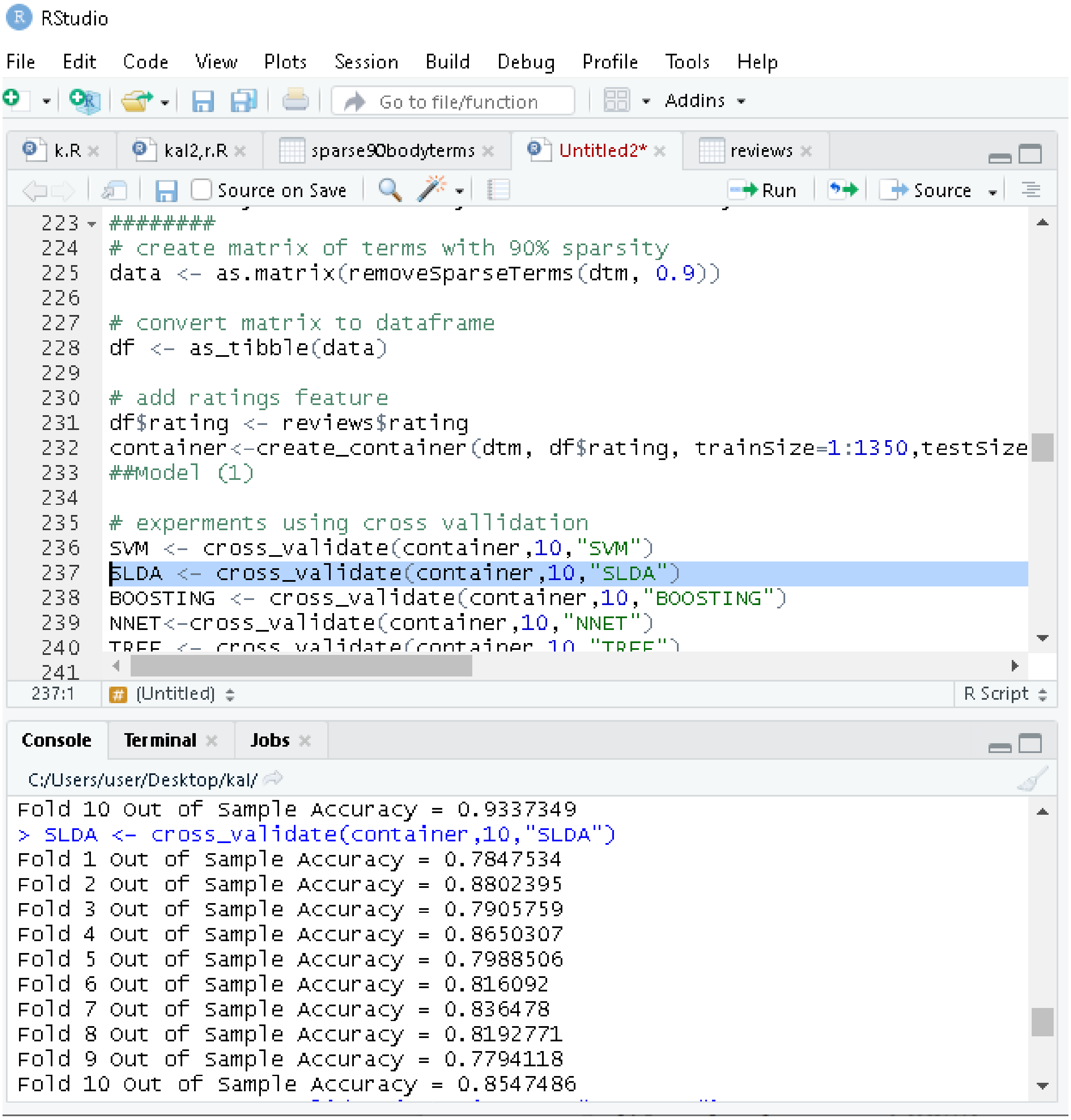
```
> # experments using cross vallidation
> SVM <- cross_validate(container,10,"SVM")
Fold 1 Out of Sample Accuracy = 0.8735632
Fold 2 Out of Sample Accuracy = 0.8777778
Fold 3 Out of Sample Accuracy = 0.8691099
Fold 4 Out of Sample Accuracy = 0.8848168
Fold 5 Out of Sample Accuracy = 0.8707865
Fold 6 Out of Sample Accuracy = 0.8888889
Fold 7 Out of Sample Accuracy = 0.9090909
Fold 8 Out of Sample Accuracy = 0.8901734
Fold 9 Out of Sample Accuracy = 0.8932584
Fold 10 Out of Sample Accuracy = 0.9337349
```

Figure 10 experiment one

The experiment above which done with SVM algorithm shows a mean accuracy of 0.933 with 10 fold cross validation and this shows the experiment is promising and valid for the given data set.

Experiment 2

The second experiment is done using SLDA classifier with the same instances and attributes like experiment one and using 10 fold cross validation.



```
########
# create matrix of terms with 90% sparsity
data <- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##Model (1)

# experiments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
```

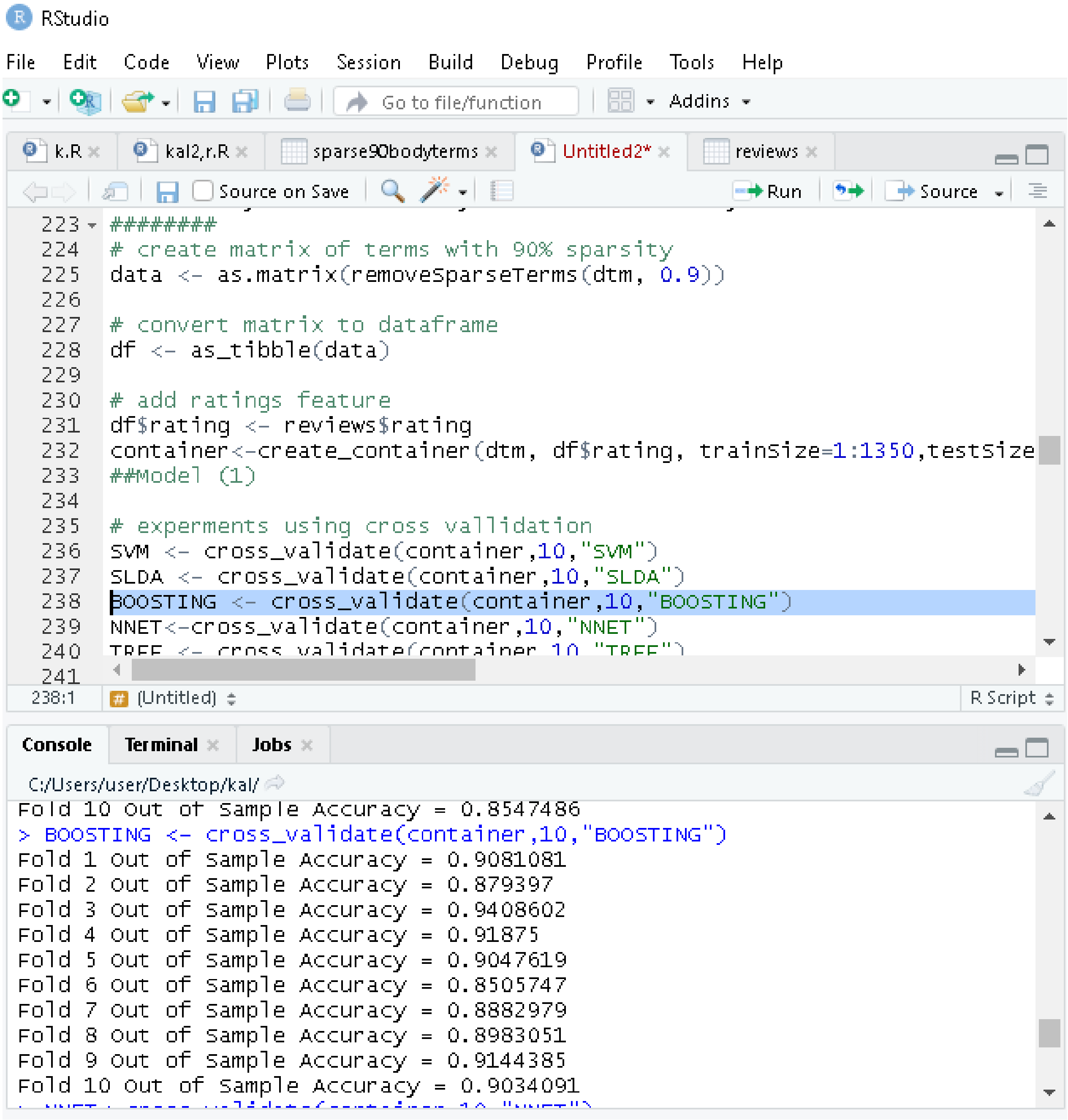
```
Fold 10 Out of Sample Accuracy = 0.9337349
> SLDA <- cross_validate(container,10,"SLDA")
Fold 1 Out of Sample Accuracy = 0.7847534
Fold 2 Out of Sample Accuracy = 0.8802395
Fold 3 Out of Sample Accuracy = 0.7905759
Fold 4 Out of Sample Accuracy = 0.8650307
Fold 5 Out of Sample Accuracy = 0.7988506
Fold 6 Out of Sample Accuracy = 0.816092
Fold 7 Out of Sample Accuracy = 0.836478
Fold 8 Out of Sample Accuracy = 0.8192771
Fold 9 Out of Sample Accuracy = 0.7794118
Fold 10 Out of Sample Accuracy = 0.8547486
```

Figure 11 experiment two

The experiment done with SLDA algorithm shows a mean accuracy of 0.88 with 10 fold cross validation and as we can see the outputs the accuracy is around 88% with error rate of 12%.

Experiment 3

The third experiment is done using BOOSTING classifier with the same instances and attributes like the above experiment and using 10 fold cross validation.



```r
########
# create matrix of terms with 90% sparsity
data <- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##Model (1)

# experiments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
```

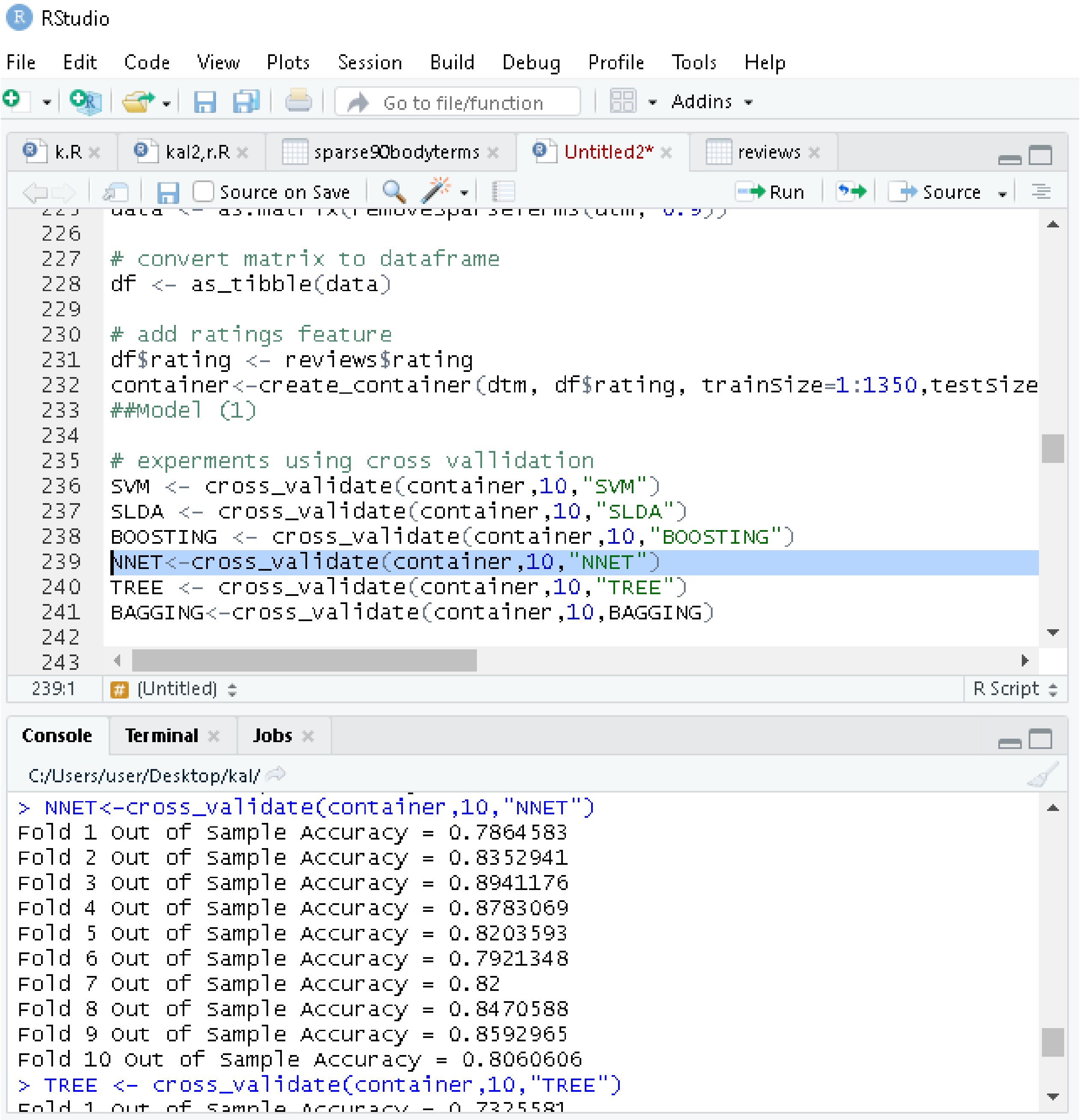
```
Fold 10 Out of Sample Accuracy = 0.8547486
> BOOSTING <- cross_validate(container,10,"BOOSTING")
Fold 1 Out of Sample Accuracy = 0.9081081
Fold 2 Out of Sample Accuracy = 0.879397
Fold 3 Out of Sample Accuracy = 0.9408602
Fold 4 Out of Sample Accuracy = 0.91875
Fold 5 Out of Sample Accuracy = 0.9047619
Fold 6 Out of Sample Accuracy = 0.8505747
Fold 7 Out of Sample Accuracy = 0.8882979
Fold 8 Out of Sample Accuracy = 0.8983051
Fold 9 Out of Sample Accuracy = 0.9144385
Fold 10 Out of Sample Accuracy = 0.9034091
> NNET <- cross_validate(container,10,"NNET")
```

Figure 12 experiment three

The experiment which is done with BOOSTING algorithm shows a mean accuracy of 0.948 with 10-fold cross validation. As the outputs of this algorithm at each fold are the same and closer to each other, the algorithm is valid for the given data set.

Experiment 4

This experiment is done using NEURAL NETWORK classifier with the same instances and attributes like the above experiment and using 10 fold cross validation.



```
data <- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##Model (1)

# experiments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
BAGGING<-cross_validate(container,10,BAGGING)
```

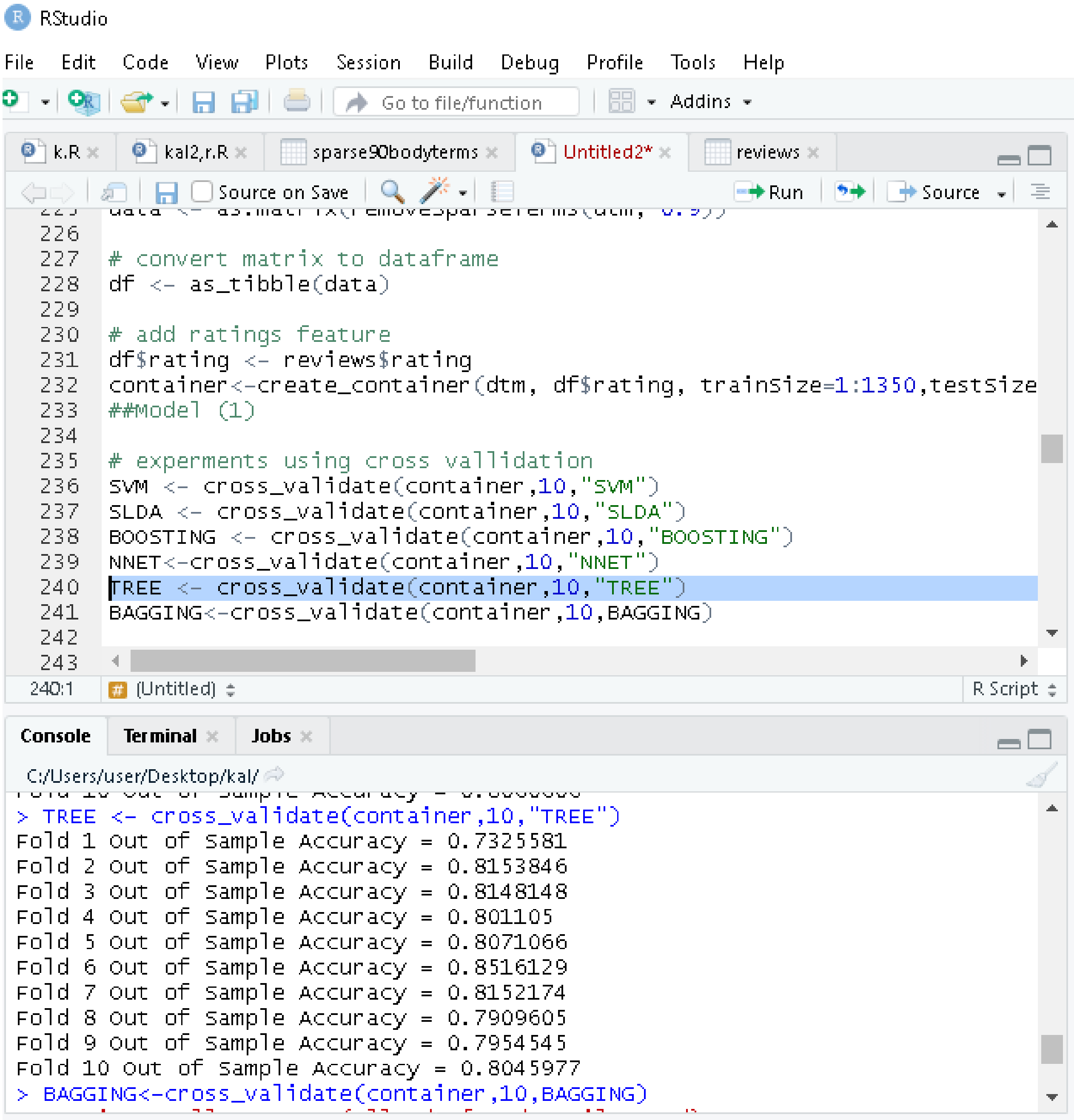
```
> NNET<-cross_validate(container,10,"NNET")
Fold 1 Out of Sample Accuracy = 0.7864583
Fold 2 Out of Sample Accuracy = 0.8352941
Fold 3 Out of Sample Accuracy = 0.8941176
Fold 4 Out of Sample Accuracy = 0.8783069
Fold 5 Out of Sample Accuracy = 0.8203593
Fold 6 Out of Sample Accuracy = 0.7921348
Fold 7 Out of Sample Accuracy = 0.82
Fold 8 Out of Sample Accuracy = 0.8470588
Fold 9 Out of Sample Accuracy = 0.8592965
Fold 10 Out of Sample Accuracy = 0.8060606
> TREE <- cross_validate(container,10,"TREE")
Fold 1 Out of Sample Accuracy = 0.7325581
```

Figure 13 experiment four

The experiment which is done with NNET algorism shows a mean accuracy of 0.894 with 10-fold cross validation. As the outputs of this algorithm at each fold are the same and closer to each other, this algorithm is also valid for the given data set.

Experiment 5

This experiment is done using TREE classifier with the same instances and attributes like the above experiment and using 10 fold cross validation.



```
data <- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##Model (1)

# experments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
BAGGING<-cross_validate(container,10,BAGGING)
```

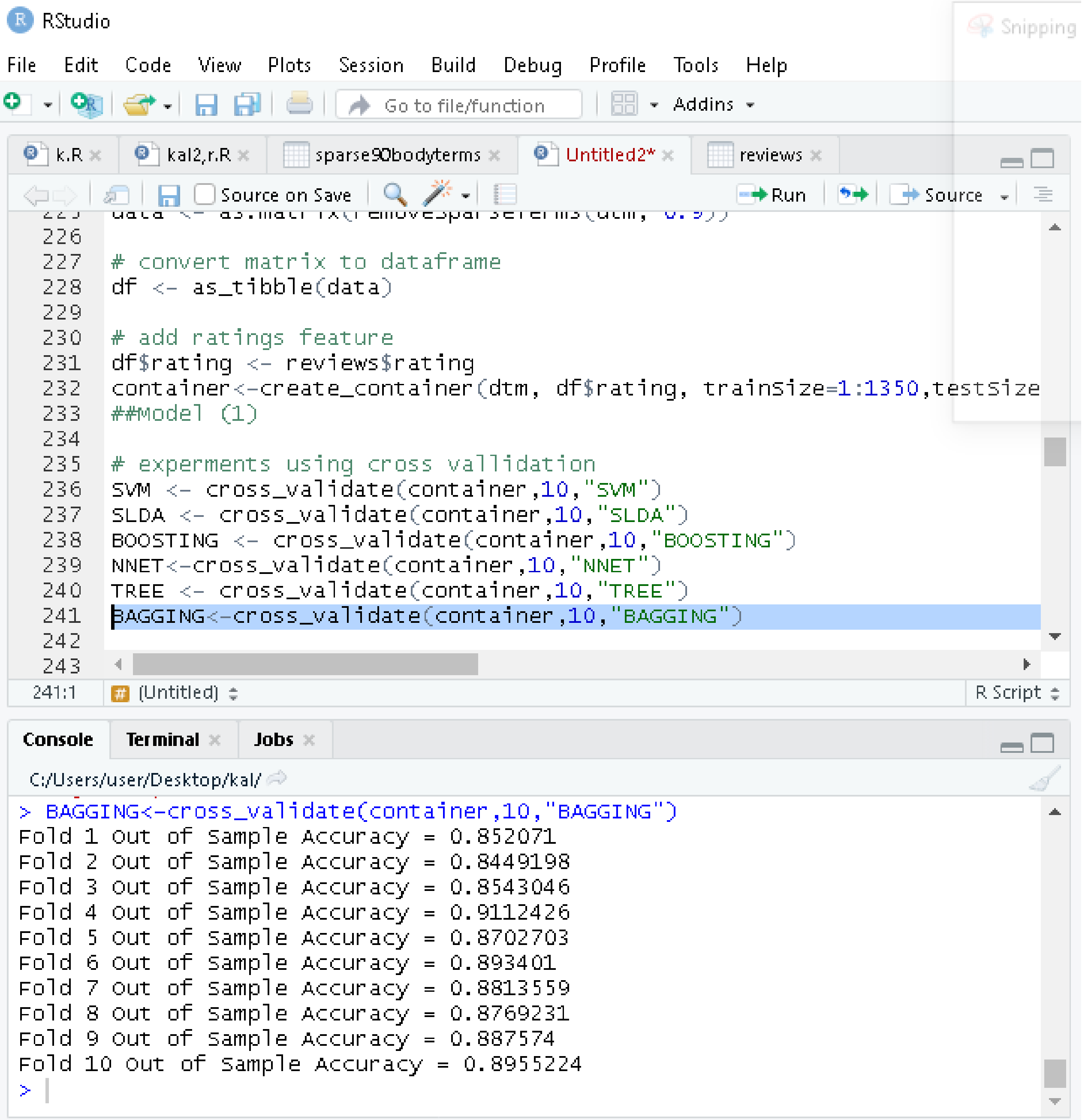
```
> TREE <- cross_validate(container,10,"TREE")
Fold 1 Out of Sample Accuracy = 0.7325581
Fold 2 Out of Sample Accuracy = 0.8153846
Fold 3 Out of Sample Accuracy = 0.8148148
Fold 4 Out of Sample Accuracy = 0.801105
Fold 5 Out of Sample Accuracy = 0.8071066
Fold 6 Out of Sample Accuracy = 0.8516129
Fold 7 Out of Sample Accuracy = 0.8152174
Fold 8 Out of Sample Accuracy = 0.7909605
Fold 9 Out of Sample Accuracy = 0.7954545
Fold 10 Out of Sample Accuracy = 0.8045977
> BAGGING<-cross_validate(container,10,BAGGING)
```

Figure 14 experiment five

The experiment done with TREE algorithm shows a mean accuracy of 0.851 with 10 fold cross validation experiment. As the outputs of this algorithm at each fold are the same and closer to each other, this algorithm is also valid for the given data set.

Experiment 6

The sixth experiment is done using BAGGING classifier with the same instances and attributes like the above experiments and using 10 fold cross validation.



```
data <- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df <- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize
##model (1)

# experments using cross vallidation
SVM <- cross_validate(container,10,"SVM")
SLDA <- cross_validate(container,10,"SLDA")
BOOSTING <- cross_validate(container,10,"BOOSTING")
NNET<-cross_validate(container,10,"NNET")
TREE <- cross_validate(container,10,"TREE")
BAGGING<-cross_validate(container,10,"BAGGING")
```

```
> BAGGING<-cross_validate(container,10,"BAGGING")
Fold 1 Out of Sample Accuracy = 0.852071
Fold 2 Out of Sample Accuracy = 0.8449198
Fold 3 Out of Sample Accuracy = 0.8543046
Fold 4 Out of Sample Accuracy = 0.9112426
Fold 5 Out of Sample Accuracy = 0.8702703
Fold 6 Out of Sample Accuracy = 0.893401
Fold 7 Out of Sample Accuracy = 0.8813559
Fold 8 Out of Sample Accuracy = 0.8769231
Fold 9 Out of Sample Accuracy = 0.887574
Fold 10 Out of Sample Accuracy = 0.8955224
>
```

Figure 15 experiment six

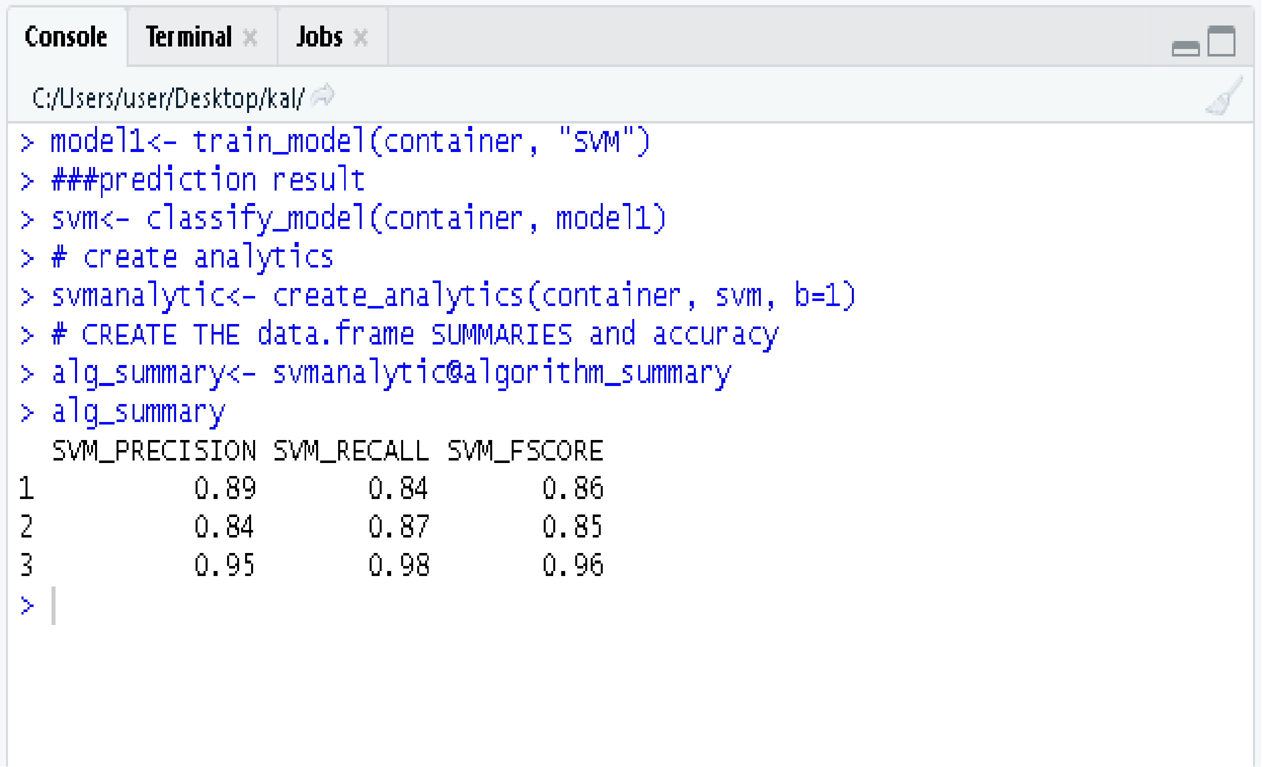
The experiment done with BAGGING algorithm shows a mean accuracy of 91% with 10 folds cross validation experiment.

Phase Two

In this second phase, models which have better and promising results are selected and experimentation is conducted with 75/25 percentage split. The algorism uses 75% of the data set for training and the remaining 25% for testing. The selected models which have better promising results include SVM classifiers, BOOSTING classifier, BAGGING and NNET.

Phase Two Experiment one

This experimentation is done using SVM classifier using the 75/25 percentage split. The model trained on the 75% of the data and test on the remaining 25%.



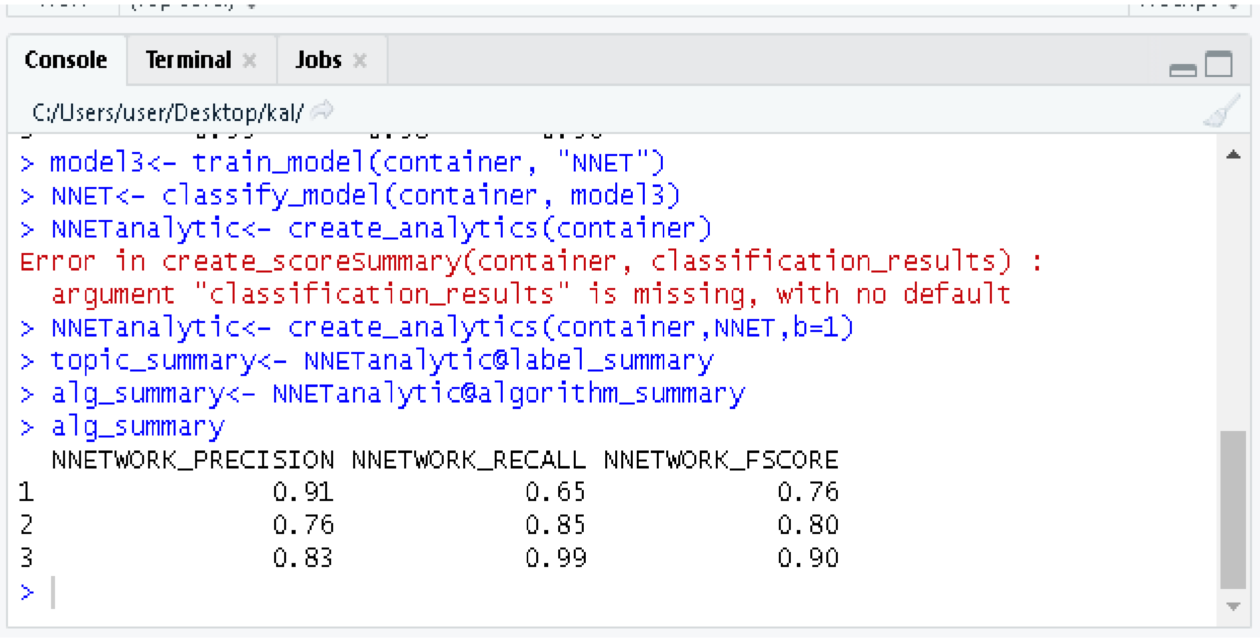
```
> model1<- train_model(container, "SVM")
> ###prediction result
> svm<- classify_model(container, model1)
> # create analytics
> svmanalytic<- create_analytics(container, svm, b=1)
> # CREATE THE data.frame SUMMARIES and accuracy
> alg_summary<- svmanalytic@algorithm_summary
> alg_summary
  SVM_PRECISION SVM_RECALL SVM_FSCORE
1          0.89       0.84       0.86
2          0.84       0.87       0.85
3          0.95       0.98       0.96
> 
```

Figure 16 Phase Two Experiment one

As the figure above shows 89.5% of predicted instances are correctly classified. Looking at the values of the FSCORE, the difference between precision and recall are minimum for the rating, moreover the FSCORE values are high. Therefore, the overall performance of the algorithm is good and this is because FSCORE determines how closer precision and recall are to each other regardless of the magnitude of the harmonic mean.

Phase Two Experiment Two

The two experiments is done using NNET classifier with 1800 instances and three class labels on 75/25 percentage split.



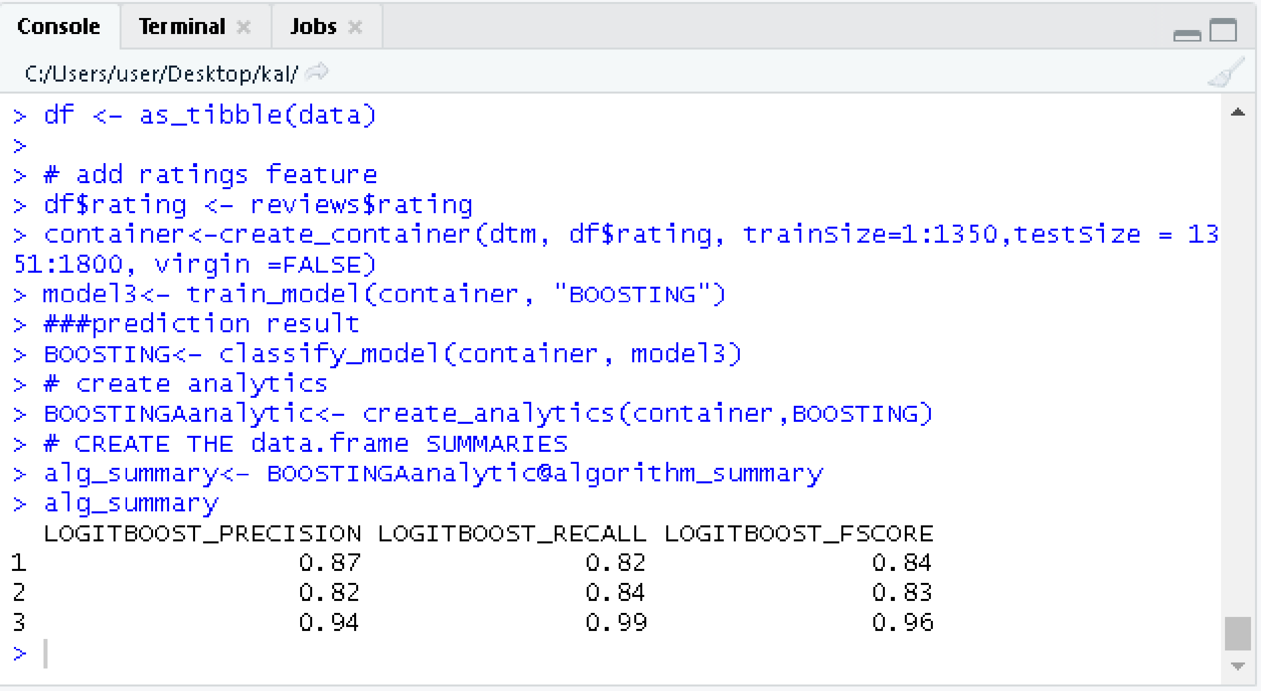
```
> model3<- train_model(container, "NNET")
> NNET<- classify_model(container, model3)
> NNETanalytic<- create_analytics(container)
Error in create_scoreSummary(container, classification_results) : 
  argument "classification_results" is missing, with no default
> NNETanalytic<- create_analytics(container,NNET,b=1)
> topic_summary<- NNETanalytic@label_summary
> alg_summary<- NNETanalytic@algorithm_summary
> alg_summary
  NNETWORK_PRECISION NNETWORK_RECALL NNETWORK_FSCORE
1               0.91            0.65            0.76
2               0.76            0.85            0.80
3               0.83            0.99            0.90
> 
```

Figure 17 Phase Two Experiment Two

As the figure above shows 83.4% of predicted instances are correctly classified. Looking at the values of the FSCORE, the difference between precision and recall are minimum for the rating, moreover the FSCORE values are high. Therefore, the overall performance of the algorithm is good and this is because FSCORE determines how closer precision and recall are to each other regardless of the magnitude of the harmonic mean.

Phase Two Experiment three

The third experiment is done using Boosting classifier with 1800 instances and three class labels on 75/25 percentage split.



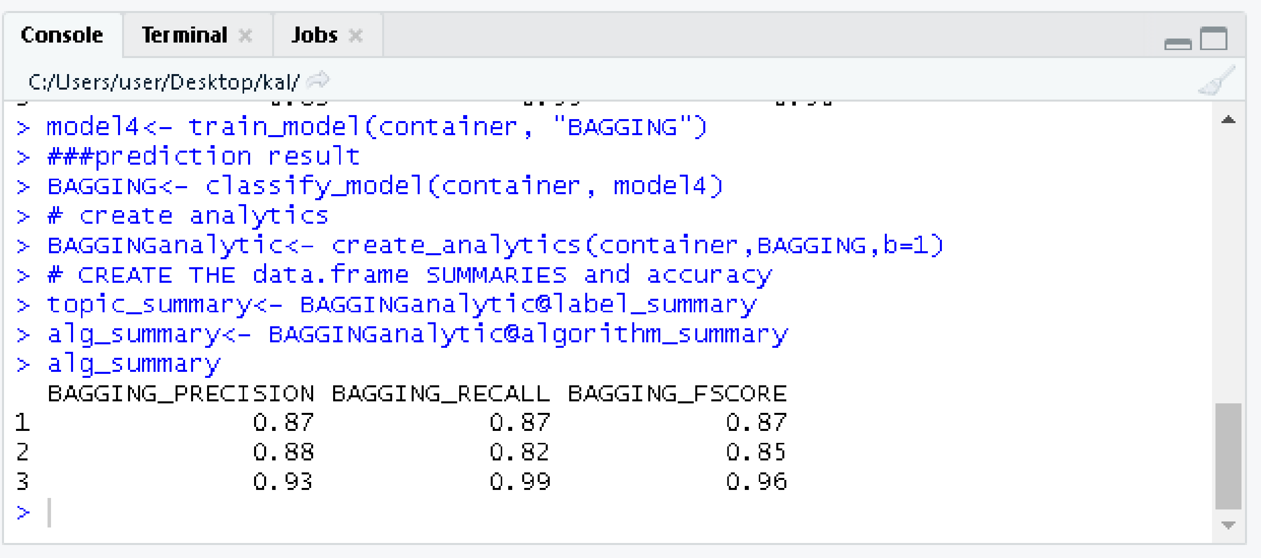
```
> df <- as_tibble(data)
>
> # add ratings feature
> df$rating <- reviews$rating
> container<-create_container(dtm, df$rating, trainSize=1:1350,testSize = 13
51:1800, virgin =FALSE)
> model3<- train_model(container, "BOOSTING")
> ###prediction result
> BOOSTING<- classify_model(container, model3)
> # create analytics
> BOOSTINGAanalytic<- create_analytics(container,BOOSTING)
> # CREATE THE data.frame SUMMARIES
> alg_summary<- BOOSTINGAanalytic@algorithm_summary
> alg_summary
  LOGITBOOST_PRECISION LOGITBOOST_RECALL LOGITBOOST_FSCORE
1                 0.87              0.82              0.84
2                 0.82              0.84              0.83
3                 0.94              0.99              0.96
> |
```

Figure 18 Phase Two Experiment theree

As the figure above shows 87.7% of predicted instances are correctly classified. Looking at the values of the FSCORE, the difference between precision and recall are minimum for the rating, moreover the FSCORE values are high. Therefore, the overall performance of the algorithm is good and this is because FSCORE determines how closer precision and recall are to each other regardless of the magnitude of the harmonic mean.

Phase Two Experiment Four

The forth experiment is done using BAGGING classifier with 1800 instances and three class labels on 75/25 percentage split.



```
> model4<- train_model(container, "BAGGING")
> ###prediction result
> BAGGING<- classify_model(container, model4)
> # create analytics
> BAGGINGanalytic<- create_analytics(container,BAGGING,b=1)
> # CREATE THE data.frame SUMMARIES and accuracy
> topic_summary<- BAGGINGanalytic@label_summary
> alg_summary<- BAGGINGanalytic@algorithm_summary
> alg_summary
  BAGGING_PRECISION BAGGING_RECALL BAGGING_FSCORE
1              0.87           0.87           0.87
2              0.88           0.82           0.85
3              0.93           0.99           0.96
> 
```

Figure 19Phase Two Experiment Four

As the figure above shows 89.4% of predicted instances are correctly classified. Looking at the values of the FSCORE, the difference between precision and recall are minimum for the rating, moreover the FSCORE values are high. Therefore, the overall performance of the algorithm is good and this is because FSCORE determines how closer precision and recall are to each other regardless of the magnitude of the harmonic mean.

Table 5 summary of experiments

Experiments	Algorithm	Techniques	Accuracy
1.	SVM	10foldcross validation	93.3%
2.	SLDA	10foldcross validation	88%
3.	BOOSTING	10foldcross validation	94.8%
4.	NEURA NETWORK	10foldcross validation	89.4%
5.	TREE	10foldcross validation	85.1%
6.	BAGGING	10foldcross validation	91%
7.	SVM	75/25 percentage split	89.5%
8.	NNET	75/25 percentage split	88%
9.	BOOSTING	75/25 percentage split	87.7%
10.	BAGGING	75/25 percentage split	89.4%

Comparing the performance of the selected four (SVM, SLDA, NNET, BAGGING) algorithm SVM performs best so that we select it as the best classifier for our project.

5.4. Evaluation of Models

Evaluation is a key point in any data mining process. It serves two purposes: the prediction of how well the final model worked in the future (or even whether it should be used at all), and as an integral part of many learning methods, which help find the model that best represents the training data [21].

The steps followed for comparison and selection are:

1. I used 10 fold cross validation techniques for six algorithms
2. Took three of the algorithms which are valid for the given data set (i.e. for each fold the sample accuracy is closer to each other and which have better mean accuracy)
3. Select one best algorithm i.e. SVM due to its better accuracy and optimal FSCORE.

5.5. Validation of the model

To test our model on actual data we conduct one experiment using new data. For this purpose we prepare 10 instances of customer reviews and checks if our model developed predicts the instance to predefined class labels. We observe that the model is validated.

5.6. Result and Discussion

The main objective of this research is to predict user rating, usefulness of review To study the domain and achieve the objective a significant tool out of different opinion mining too to collect and conduct experiment has been identified. Finally a predictive model which predicts user ratings is built. The results achieved by applying the selected data mining algorithm (BOOSTING) for classification on the collected data reveal that our model has an overall accuracy of 94.8%. The classifiers that we adopted in this work are: SVM, BOOSTING, SLDA, NNETWOR, TREE and BAGGING. As each algorithm uses different parameters and techniques to learn from the training data, to predict a new data we did several experiments. In order to find the optimal classifier which correctly classifies our data, all experiments were performed with cross-validation and make sure that the parameters are not optimized for one particular test set and performed the experiment using 75/25 percentage split of the data sets by doing 10-fold cross validation experiment selected (SVM, BOOSTING and BAGGING) which work on full data set. Finally, according to the criteria stated in section 3 of the performance evaluation technique, BOOSTING and SVM is selected as best algorithm developed a user opinion rating (positive, negative, neutral) prediction model. The research questions of this study were:

1. Is it possible to extract customer opinions on web (social media) and analyze?
2. How can we analyze, classify customer feedback from opinion collected from web (social media) and evaluate?

It is important to notice that most of the current research on the application of data mining is primarily used for applications such as classifying email spam and network intrusion detection. Text classification for sentiment analysis and product review has not been given much attention by researchers. Regarding to the first research question, is it possible to extract customer opinions about a product on social media and rate them as 1, 2, 3? As can be stated on section three customer opinions are easily extracted from Facebook using Facebook developer API and R package. We have tried to collect data from Tecno mobile Facebook page and developed a model that can rate mobile product based on the customer opinion. The second research question,how can we analyze, classify customer feedback from opinion collected from web (social media) and evaluate? It is shown that prediction of customer opinions on product reviews is possible by supervised machine learning algorithm once user opinions are clearly defined and labeled(rated as 1,2,3) correctly by experts.

5.7. Interface

A typical workflow for building an advanced analytics solution starts with data exploration and predictive modeling, develops R scripts and models that prove effective for the task at hand. After the scripts and models are ready they can be deployed into production and integrated with existing or new applications. To develop a full application we use the following tools.

- SQL Server R services
- R.NET
- Visual studio

5.8. Development

We use R to explore data and build predictive models from the workstation using an R IDE of our choosing. R Services (In-database) client components provide us with all the tools needed to experiment and develop. These tools include the R runtime, the Intel math kernel library to boost the performance of standard R operations, and a set of enhanced R packages that support executing R code in SQL Server. Then we can connect to SQL Server and bring the data to the client for local analysis, as usual. However, a better solution is to use the ScaleR APIs to push computations to the SQL Server computer, avoiding costly and insecure data movement. To develop R solutions, we can use any Windows-based IDE that supports R, including R Tools for Visual Studio

Chapter six
Conclusion and recommendation

6.1. Conclusions

The growing use of opinion on social media which needs text mining, machine learning, natural language processing techniques and methodologies to organize and extract pattern and classifying user opinion. This research focused on the existing literature and explored product reviews and an analysis of customer opinion given on product extracted from Tecno Facebook page and method to rate product reviews classify text documents. We presented a method that can automatically rate product reviews. We showed a way to extract customer opinion from Facebook and define features that analyze the content of messages, such as positive opinion, neutral opinion and negative opinion. These features are trained on customer opinion on product that was selected with a short list of query keywords. This list can easily be modified and extended to refine the existing features or to define new categories for another domain. The grammar and vocabulary used in a sentence separates the type of activities. Also sentiment of the word on the product reviews is used to train the machine. In combination with our post processing steps, we are able to rate a product that have a great importance for the company to improve their product quality and serve their customer efficiently. We experimentally tested our approach using datasets that we collected from Tecno Facebook page. The datasets were collected using the Facebook developer API. We converted the datasets into a relational database to make it easier to process data and extract features. This paper also gives a brief introduction to the various text pre-processing and classifier algorithms. The existing classification methods are compared and contrasted based on their accuracy. It was verified from the research that customer opinion can help a company to improve their product and compute with other company. Also this research project shows machine learning algorithms classifies text document accurately if properly labeled and adding additional features with the feature we used in the project. From the above discussion it is understood that different algorithms perform differently depending on data collection. However, to the certain extent SVM and BOOSTING with term weighted representation scheme performs well in opinion classification tasks. Future work will focus on the development of an interactive demonstrator to study the effects on overall performance for the user, to assess the strong and weak points in this system, and to get feedback from our stakeholders.

6.2. Future Work and recommendations

Another extension to our work would be to implement some feature engineering methods such as feature extraction to see if more efficient and accurate classifies can be trained. Also techniques such as query expansion can be applied to our problem to exploit additional auxiliary information, labeling of the text and developing a dictionary of a threat word to improve the performance of our classifiers.

In fact other features from comment of mobile in Facebook can be extracted and used as additional features and site to improve the performance of classification. Furthermore, more advanced classifiers such as deep neural networks can be employed for this problem to see if they are suitable for this task or not.

REFERENCE

[1] Mikalai, T., & Themis, P. (2012). Survey on Mining Subjective Data on the Web. *Data Mining and Knowledge Discovery,* DOI: 10.1007/s10618-01100238-6.

[2] WalaaMedhat, Ahmed Hassan, HodaKorashy, and Sentiment Analysis: A Survey, Ain shams Engineering Journal, vol.5, pp. 1093-1113, (2014).

[3]. Sarwar Shah Khan, Muzammil Khan, Qiong Ran and Rashid Naseem. "Challenges in Opinion Mining, Comprehensive Review". A Science and Technology Journal (Ciencia e TecnicaVitivinicola). Vol. 33, no. 11, pp. 123-135. 2018

[4]. Kaur, A., & Gupta, V. “A survey on sentiment analysis and opinion mining techniques”. Journal of Emerging Technologies in Web Intelligence, 5(4), 367-371.2013

[5]. Yi-ChingZeng, Tsun Ku, Shih-Hung Wu, “Modeling the Helpful Opinion Mining of Online Consumer Reviews as a Classification Problem”. Computational Linguistics and Chinese Language ProcessingVol.19, No. 2. 2014

[6]. Sharma, R., Nigam, S., & Jain, R. “Opinion mining in Hindi language: a survey”. arXiv preprint arXiv:1404.4935.2014

[7]. BakhtawarSeerat, FarouqueAzam, “Opinion Mining: Issues and Challenges (A survey),” International Journal of Computer Applications (0975 – 8887) Volume 49– No.9. 2012

[8]. Surya Prakash Sharma1, DrRajdev Tiwari2, Dr Rajesh Prasad3, “Opinion Mining and Sentiment Analysis on Customer Review Documents- A Survey”. Research challenge on opinion mining and sentiment analysis. International Conference on Advances in Computational Techniques and Research Practices Noida Institute of Engineering & Technology, Greater Noida Vol. 6, Special Issue 2. 2017

[9]. Tulu Tilahun, “Opinion Mining from Amharic Blog”. Addis Ababa university.2013

[10]. Nidhi R. Sharma, “Opinion Mining, Analysis and its Challenges”. International Journal of Innovations & Advancement in Computer Science IJIACS, ISSN 2347 – 8616, Volume 3, Issue 1, 2014

[11]. Aminu Mohammed, “Design and Implementation of SMS Based Public Opinion Polling System”. Addis Ababa University, 2010

[12]. N. M. Shelke, S. Deshpande and V. Thakre “Survey of Techniques for Opinion Mining” International Journal of Computer Applications (0975 – 8887) Volume 57– No.13, November 2012.

[13] Bing Liu, “Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers”, 2012

[14] AlemuMolla, “Identifying Negative Deception Using Opinion Mining Techniques in Twitter” (2010)

[15] https://www.yotpo.com,” Find out how customer reviews can help boost your brand”

[16] GangWang, “Sentiment classification: The contribution of ensemble learning”, Decision Support Systems Volume 57, 2014,

[17] AlessiaD’Andrea, “Approaches, Tools and Applications for Sentiment Analysis Implementation”, International Journal of Computer Applications Volume 125 – No.3, 2015

[18] Dongjoo Lee and Ok-Ran Jeong, “Opinion Mining of Customer Feedback Data on the Web”

[19] Abulaish , “Feature and Opinion Mining for Customer Review Summarization”,2017

[20] ZerihunTolla, “Threat prediction based on opinion extracted from twitter”, HiLCoE
School of Computer Science & Technology, 2016

[21] Agar waletal, “Sentiment Analysis of Twitter Data”, Columbia University New York 2017

[22] Strauss, A. and J. Corbin, “Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory”, CA, Thousand Oaks: Sage Publications, 1998.

[23] KjerstiAas and Line Eikvil “Text Categorization: A Survey” Report No. 941. ISBN 82-539-0425-8., June, 1999.

[24] Rob Schapire “Machine Learning Algorithms for Classification”, Princeton University

[25] Alex Smola and S.V.N. Vishwanathan“Introduction to Machine Learning” Cambridge University Press 2008

[26] Richards, J. “Remote Sensing Digital Image Analysis. New York”, Springer, (2013).

[27] Kontostathis, A., Edwards, L., Leatherman, A., “Text mining and cybercrime”, Chapter 8 in Text mining – Applications and Theory, (2010).

[28] V O Tama et al “Labeling Analysis in the Classification of Product Review Sentiments by using Multinomial Naive Bayes Algorithm” The 2nd International Conference on Data and Information Science (2019).

[29] ZeeniaSingla, “Sentiment Analysis of Customer Product Reviews Using Machine Learning”, International Conference on Intelligent Computing and Control, (2017)

[30] Arif Abdurrahman Farisi et al “Sentiment analysis on hotel reviews using Multinomial Naïve Bayes classifier” Journal of Physics,(2019)

[31] Minu P Abraham, “Feature Based Sentiment Analysis of Mobile Product Reviews using Machine Learning Techniques” Volume 9 No.2, March - April (2020)

[32] SepidehPaknejad, “Sentiment classification on Amazon reviews using machine learning approaches” (2018)

[33] M. S. Neethu and R. Rajasree, "Sentiment analysis in twitter using machine learning techniques,"

2013 Fourth International Conference on Computing, Communications and Networking Technologies, 2013. https://doi.org/10.1109/ICCCNT.2013.6726818

[34] AnuragMulkalwar, KavitaKelkar Sentiment “Analysis on Movie Reviews Based on Combined Approach”, International Journal of Science and Research, Volume 3 Issue 7, July 2014.

[35] Pete Chapman (NCR), Julian Clinton (SPSS) “Step-by-step data mining guide”, book publish 2010.

[36] Richard G. Brereton and Gavin R. Lloy,“Support Vector Machines for classification and regression”,www.rsc.org/analyst, December 2019

[37]Guoqiang Peter Zhang “Neural Networks for Classification: A Survey”,IEEE transactions on systems, Nov 2013

[38] M. Bianchini, F. Scarselli, “Neural Networks and Learning Systems”IEEEComputer Science, Published 2014

[39] Pavan Vadapalli “Bagging vs Boosting in Machine Learning: Difference betweenBagging and Boosting”, www.upgrad.com, Nov 12, 2020

[40]John Duchi“Supplemental Lecture notes” http://cs229.stanford.edu/,Jun 2017

[41]David M. Blei and Jon D. McAuliffe “Supervised topic models”Princeton University Princeton, NJ,2008

[42] www.analyticsvidhya.com“Tree Based Algorithms: A Complete Tutorial from Scratch (in R & Python)”, APRIL 12, 2016

[43]lan Witten,“The WEKA data mining software: An update”The University of Waikato,November 2009

[44]Svetlana S. Aksenova “Machine Learning with WEKA, WEKA Explorer Tutorial”California State University,2009

[45] Susan L. Miertschin, “A Data Mining Tool”,https://www.cs.waikato.ac.nz/ml/weka/

Appendix

Rcode

```
# Load libraries
library(tm)
library(SparseM)
library(RTextTools)
library(e1071)
library(doParallel)
library(caret)

# Find how many cores are on your machine
#detectCores() # Result = typically 4-8

# Create Cluster with desired number of cores. Don't use them all! Your computer is running other processes.
cl<- makeCluster(4)

# Register Cluster
registerDoParallel(cl)

# Confirm how many cores are now "assigned" to R and RStudio
#getDoParWorkers() # Result 4

# Stop Cluster. After performing tasks, stop your cluster.
stopCluster(cl)
##### load library
# libraries
if (require(pacman) == FALSE) {
install.packages("pacman")
}
pacman::p_load(
tidyverse, magrittr,
  # preprocessing
lubridate,
  # html tables creation
knitr, kableExtra,
  # visualization tools
gridExtra,
  # text mining
tm, SnowballC, wordcloud,
  # machine learning
caret, modelr
)
###### define them for plots
# define themes for the plots
theme_set(theme_bw())
###### explore data
setwd("C:/Users/user/Desktop/kal/")
reviews<- read.csv("reviews.csv")
validate<- read.csv("validate.csv")
set.seed(42)
View(reviews)
##write.csv(reviews,"C:\\Users\\user\\Desktop\\MyData.csv", row.names = FALSE)

glimpse(reviews)
###1st Iteration

###Iteration goal: Text mining with review body text. ## Preprocessing (1)
###Change data types

reviews$name <- as.character(reviews$name)
reviews$rating <- as.factor(reviews$rating)
reviews$date <- as.character(reviews$date)
reviews$body <- as.character(reviews$body)
reviews$title <- as.character(reviews$title)
reviews$asin <- as.character(reviews$asin)
```

```
reviews$Date <- as.Date(reviews$date, "%B %e, %Y")

#### Remove observations with missing reviews

is.na(reviews$body) %>% summary()
reviews[!complete.cases(reviews$body), ] # rows w missing review

reviews<- reviews[complete.cases(reviews$body), ]

####Select only the reviews (body) and rating (scale of 1-5 stars).

knitr::kable(reviews %>%
select(rating, body) %>%
head(2)) %>%
  kableExtra::kable_styling(full_width = F, position = "center")

####rating      body
###Text mining (1): body text

###Import data

reviewtext<- VCorpus(VectorSource(reviews$body))

##Inspect Corpora
print(reviewtext)
inspect(reviewtext[1:2])
lapply(reviewtext[1:2], as.character)

####Cleanse text

# -- Eliminate Extra Whitespace
reviewtext<- tm_map(reviewtext, stripWhitespace)
```

```
# -- Convert to Lower Case
reviewtext<- tm_map(reviewtext, content_transformer(tolower))

# -- Remove Stopwords
reviewtext<- tm_map(reviewtext, removeWords, stopwords("english"))

# -- Remove Punctuation
reviewtext<- tm_map(reviewtext, removePunctuation)

# -- Remove Numbers
reviewtext<- tm_map(reviewtext, removeNumbers)

# -- Stemming
reviewtext<- tm_map(reviewtext, stemDocument)

####Create DocumentTermMatrix

dtm<- DocumentTermMatrix(reviewtext) #documents as rows, terms as columns
inspect(dtm)
###Explore Matrix ----Find terms that occur 6000+ times

freqtermsbody<- findFreqTerms(dtm, 6000)
freqtermsbody

###Find associations (i.e. terms which correlate) with minimum 50% correlation for the term “phone”

assocbody<- findAssocs(dtm, "phone", 0.5)
assocbody

###Remove terms which have at least 90% of sparse elements (i.e., terms occurring in only 90%+ of text)
```

```
sparse90bodyterms <- as.matrix(removeSparseTerms(dtm, 0.9))
sparse90bodyterms %>% colnames()

###6. Explore Dictionary This is useful for restricting the dimension of the matrix a priori + focus on specific term for distinct text mining contexts

dictsummarybody<- summary(as.matrix(DocumentTermMatrix(reviewtext,
      list(dictionary = c("phone", "screen", "battery", "camera", "price")))))
dictsummarybody

### 7. Explore Frequency

# -- Count
freq<- sort(colSums(as.matrix(removeSparseTerms(dtm, 0.95))), decreasing = T)

# -- Visualize WordCloud
wordcloud(names(freq), freq,scale = c(6,0.5), random.order = FALSE, colors = brewer.pal(8, "Dark2"))

##Prepare data for modeling (1)

###Create a dataframe that includes terms with 90% sparsity – i.e. terms that appear in at least 90% of all reviews (body text)

# create matrix of terms with 90% sparsity
data<- as.matrix(removeSparseTerms(dtm, 0.9))

# convert matrix to dataframe
df<- as_tibble(data)

# add ratings feature
df$rating <- reviews$rating
```

```
container<-create_container(dtm, df$rating, trainSize=1:1350,testSize = 1351:1800, virgin =FALSE)


##Model (1)
####train model using svm
model1<- train_model(container, "SVM")
###prediction result
svm<- classify_model(container, model1)
# create analytics
svmanalytic<- create_analytics(container, svm, b=1)
# CREATE THE data.frame SUMMARIES and accuracy
topic_summary<- svmanalytic@label_summary
alg_summary<- svmanalytic@algorithm_summary
alg_summary

model4<- train_model(container, "BAGGING")
###prediction result
BAGGING<- classify_model(container, model4)
# create analytics
BAGGINGanalytic<- create_analytics(container,BAGGING,b=1)
# CREATE THE data.frame SUMMARIES and accuracy
topic_summary<- BAGGINGanalytic@label_summary
alg_summary<- BAGGINGanalytic@algorithm_summary
alg_summary

model2<- train_model(container, "SLDA")
model3<- train_model(container, "BOOSTING")
model4<-train_model(container,"NNET")
model5<- train_model(container,"TREE")
model6<- train_model(container, "BAGGING")
###prediction result
svm<- classify_model(container, model1)
```

```
SLDA<- classify_model(container, model2)
BOOSTING<- classify_model(container, model3)
NNET<-classify_model(container,model4)
TREE<- classify_model(container, model5)
BAGGING<- classify_model(container, model6)

# create analytics
svmanalytic<- create_analytics(container, svm, b=1)
SLDAanalytic<- create_analytics(container,SLDA)
BOOSTINGAanalytic<- create_analytics(container,BOOSTING)
NNETAanalytic<- create_analytics(container,NNET)
TREEanalytic<- create_analytics(container,TREE)
BAGGINGanalytic<- create_analytics(container,BAGGING)
# CREATE THE data.frame SUMMARIES
topic_summary<- BOOSTINGAanalytic@label_summary
alg_summary<- BOOSTINGAanalytic@algorithm_summary
alg_summary
ens_summary<-svmanalytic@ensemble_summary
ens_summary
doc_summary<- svmanalytic@document_summary
recall_accuracy(svmanalytic@document_summary$MANUAL_CODE,svmanalytic@docu
          ment_summary$CONSENSUS_CODE)
###SLDA
topic_summary<- SLDAanalytic@label_summary
alg_summary2<- SLDAanalytic@algorithm_summary
alg_summary2
ens_summary<-SLDAanalytic@ensemble_summary
doc_summary<- SLDAanalytic@document_summary
###BOOSTING
topic_summary3<- BOOSTINGAanalytic@label_summary
alg_summary3<- BOOSTINGAanalytic@algorithm_summary
alg_summary3
ens_summary3<-BOOSTINGAanalytic@ensemble_summary
```

```
ens_summary3
doc_summary3<- BOOSTINGAanalytic@document_summary
###NNET
topic_summary4<- NNETAanalytic@label_summary
alg_summary4<- NNETAanalytic@algorithm_summary
alg_summary4
ens_summary4<-NNETAanalytic@ensemble_summary
doc_summary4<- NNETAanalytic@document_summary
###TREE
topic_summary5<- TREEanalytic@label_summary
alg_summary5<- TREEanalytic@algorithm_summary
alg_summary5
ens_summary5<-TREEanalytic@ensemble_summary
ens_summary5
doc_summary5<- TREEanalytic@document_summary
###BAGGING
topic_summary6<- BAGGINGanalytic@label_summary
alg_summary6<- BAGGINGanalytic@algorithm_summary
alg_summary6
ens_summary6<-BAGGINGanalytic@ensemble_summary
doc_summary6<- BAGGINGanalytic@document_summary
```