CONSUMER SENTIMENT ANALYSIS USING TWITTER

A Paper
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Rumana Rashid

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Program:
Software Engineering

April 2017

Fargo, North Dakota

North Dakota State University
Graduate School

Title

CONSUMER SENTIMENT ANALYSIS USING TWITTER

By

Rumana Rashid

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Kendall Nygard
Chair

Dr. Kenneth Magel

Dr. Dean Steele

Approved:

April 12, 2017
Date

Dr. Brian M. Slator
Department Chair

ABSTRACT

Sentiment analysis is the task of finding people's opinions about specific objects/matters. Ordinary people's opinions affect the decision-making process. Today, there is a massive explosion of "sentiments" available on social media, e.g. Twitter. Twitter is one of the most popular worldwide social-networking services. Twitter is a widely-used way to get people's opinion about some topics when you read their posts. In this study, a model was coded in R environment and implemented and tested using a large dataset to estimate people's opinions concerning specific topics that can be used for implementing better decision in market research. To achieve this goal, a large set of Twitter data along with a reference to a specific business was captured and fed to two models namely, Naïve Bayes and Support Vector Machine (SVM) to classify them. Then, I obtained the result at identifying percentage of positive and negative opinions towards the specific business.

ACKNOWLEDGEMENTS

I wish to express my deep sense of respect and indebtedness to my adviser, Dr. Nygard, for his valuable suggestions, wonderful guidance, and unending encouragement during the research project. I also would like to thank Dr. Magel and Dr. Steele for taking the time to be members of my supervisory committee. I appreciate their time, kindness, and valuable support.

Also, I would like to thank my friends, especially Abdulaziz, Eshita, Nazia and Shampa, for their inspiration and support at different stages of my work. Finally, I would like to thank my husband, Shahad; my caring parents; other family member; and Dr. Borhan and Sabiha for their inspiration, encouragement, everlasting blessings, and abundant love for me regarding the successful completion of this goal.

DEDICATION

I dedicate this dissertation to my mother, Rizia Begum, who has supported me throughout every stage of my life.

TABLE OF CONTENTS

ABSTRACT iii

ACKNOWLEDGEMENTS iv

DEDICATION v

LIST OF TABLES viii

LIST OF FIGURES ix

1. INTRODUCTION 1

2. BACKGROUND AND RELATED WORK 3

2.1. Sentiment Analysis 3

2.2. Naïve Bayes Algorithm 3

2.3. Support Vector Machine 8

2.4. Related Work 9

3. SYSTEM DESIGN 11

3.1. Capturing Twitter Feeds into R Studio 11

3.1.1. Getting Twitter API Keys 11

3.1.2. Start to Extract Tweets in R 12

3.2. Data-Vector Representation and Cleaning/Formatting 12

3.3. Sentiment Analysis 15

3.3.1. Naïve Bayes Classifier 15

3.3.2. SVM Classifier 15

3.4. Experiments with the Model 16

3.5. Precision and Recall 16

4. EXPERIMENT AND RESULT 18

4.1. Dataset 18

4.2. Experiments 18

4.2.1. Estimating Sentiment with the Naïve Bayes Method 18

4.2.2. Estimating Sentiment with Support Vector Machine 20

4.3. Overall Comparison of the Results from the Two Algorithms 22

4.4. Accuracy of the Algorithms 23

4.4.1. Accuracy of the Naïve Bayes Algorithm: Precision and Recall 23

4.4.2. Accuracy of the Support Vector Machine (SVM) Algorithm: Precision and Recall 24

4.4.3. 10-fold Cross-Validation 25

4.4.4. Coefficient of Variation 27

5. CONCLUSION 28

6. FUTURE WORK 29

6.1. Sentiment Analysis for Big Data 29

6.2. Twitter Analysis on Other Aspects 29

7. REFERENCES 30

LIST OF TABLES

Table Page

1. Example of positive and negative words from subjectivity lexicon 15

2. Example of TP, TN, FN, and FP 17

3. Results from using Naïve Bayes for the Walmart and Target experiments 19

4. Results from the experiment using SVM for the Walmart and Target Experiments 21

LIST OF FIGURES

Figure Page

1. Objects classified as either green or red 6

2. Classify new object (white circle)..... 7

3. Support Vector Machine (SVM) for classification..... 9

4. Create a new app..... 11

5. Get API keys..... 12

6. Flow for cleaning/formatting tweets..... 13

7. Sample codes for cleaning/formatting data..... 13

8. Tweets without the cleaning/formatting process 14

9. Tweets with the cleaning/formatting process 14

10. Naïve Bayes analysis for Walmart and Target 19

11. SVM analysis for Walmart and Target..... 21

12. Walmart supermarket sentiment-analysis results for the Naïve Bayes and SVM methods. 22

13. Target supermarket sentiment-analysis results for the Naïve Bayes and SVM methods 23

14. Precision-Recall for the Naive Bayes algorithm..... 24

15. Precision and Recall for SVM 25

16. 10-fold cross validation with the Walmart data..... 25

17. 10-fold cross validation for the Target supermarket data. 26

1. INTRODUCTION

Market research is an organized effort to gather information about the customers' opinion about a specific product. By doing the market research, a company maintains competitiveness over its competitors. Market research is a very important component of business strategy [1]. Market research includes social and opinion research. Market research is the systematic gathering and analysis of information about individuals or organizations by using analytical methods and techniques to obtain feedback or to support the decision making for business [2] Sentiment analysis is used as opinion research to find positivity or negativity towards a product or topic.

For a business, it is very important to efficiently determine the positivity or negativity for the public's opinion about a product. Because time is money or sometimes even more valuable than money, instead of spending time reading and figuring out a text's positivity or negativity, a business can use a sentiment analysis, coming from a social network, to determine opinions about its products. This sentiment analysis can help the business to obtain feedback about its products, and this process will help with the company's marketing decisions.

The use of electronic media is increasing daily. People share their opinions, emotions, and feelings through electronic media (e.g., Twitter or Facebook). Social networks have become a necessary part of our daily life. Among all social-network media, Twitter has become one of the most important platforms to share information and to communicate with friends. People tweet about various topics: movies, products, brands, etc. Twitter only allows people to publish 140 characters in a single tweet, making the information easy to read and to spread. The most important and valuable data in these tweets are the people's sentiments. When people post a tweet, they have feelings and attitudes, such as satisfaction or dissatisfaction, or a positive or

negative feeling, about the thing that they mentioned. These sentiment data would be a great source for companies or institutions to conduct marketing research and customer surveys. The correlation between public opinions and the company's stock price has been discussed [3]. One stock-price indicator is the market, and the customers' behavior has a significant impact on the market. Generally, the public sentiment about a company and its products is proportional to the company's stock price behavior [4].

In this project, I developed a model using R which does the sentiment analysis on Twitter feed as public opinion. For this model, I used the Naïve Bayes and Support Vector Machine (SVM) algorithms to conduct the sentiment analysis. This model could be utilized as a tool for the business' market analysis.

2. BACKGROUND AND RELATED WORK

This chapter is dedicated to covering the essential background which is important for the study. Moreover, I review the related research as a comparison to my work.

2.1. Sentiment Analysis

Sentiment analysis is the task of determining people's opinions about specific objects/matters. Ordinary people's opinions affect the decision-making process. Today, there is an enormous explosion of "sentiments" available on social media, e.g. Twitter. Text pieces from Twitter are a very good source for companies and individuals who want to observe their reputation and to obtain feedback about their products and actions. Sentiment analysis gives a company the ability to monitor different social media sites in real time and to act accordingly. Direct beneficiaries of the sentiment analysis technology are the marketing personnel, campaign managers, politicians, investors, and online shoppers [5].

A basic task of sentiment analysis is classifying the polarity of a given text in a document or a sentence in order to determine whether the opinion is positive or negative. Early work in this area was done by Turney and Pang [6] [7] who applied different methods to detect the polarity of product and movie reviews. Sentiment-analysis experiments have been done using Naive Bayes, Maximum-Entropy classifiers, and Support Vector Machines (SVM).

In this paper, I present sentiment classification based on the Naïve Bayes algorithm and the Support Vector Machine (SVM). The philosophies behind these two algorithms are quite different, but each one has been shown to be effective in previous text-categorization studies.

2.2. Naïve Bayes Algorithm

Naïve Bayes is a probabilistic model that tends to work well with text classifications. As a supervised learning method, the Naïve Bayes classifier often performs very well in practice. If

we are looking for a straightforward and good performance method, then the Naïve Bayes classifier is a good option. The Naive Bayes classifier technique is based on the Bayesian theorem, which provides a formula for estimating the probability that an item with known attributes belongs to a class.

Suppose that we have m classes that we denote as y_1, y_2,…., y_m and in the approach of supervised learning, we have a collection of items with known classifications. From these known items, let $P(y_1)$, $P(y_2)$,…., $P(y_m)$ be the proportions of the items in each class. For example, if we have 100,000 example items and exactly 10,000 of them are in class y_1, then the proportion is 0.10 (i.e. 10%). In the Naïve Bayes method, these proportions are treated as probabilities, which assumes that there are enough example items to make this a reasonable assumption. Assuming that each item has p attributes, let x_1, x_2,…., x_p be the specific set of attributes for a new item for which we don't know the class to which it belongs. The aim is to classify the item. Using the usual notation for conditional probability, we use the expression $P\{A|B\}$ to represent the probability that event A occurs, given that event B occurs. Now let X_1, X_2,….,X_p be the random variables for the predictor classes. Bayes theorem is given as follows:

$$P\{y_i|X_1, X_2, \dots, X_n\} = \frac{P\{X_1, X_2, \dots, X_n|y_i\}P\{y_i\}}{P\{X_1, X_2, \dots, X_n|y_1\}P\{y_1\} + \cdots . + P\{X_1, X_2, \dots, X_n|y_m\}P\{y_m\}}$$

This is called the posterior probability of belonging in class y_i, since the expression includes the predictor information. $P(y_i)$ is called the prior probability, known in advance of any information about the attributes.

The right-hand of Bayes formula involves two types of information. (1) The probabilities $P\{y_i\}$, which are estimated by simply using the proportions obtained by counting the occurrences of each class in the data and, (2) the conditional probabilities $P\{X_1, X_2, \dots, X_n|y_i\}$. This expression contains a joint probability, meaning that X_1 occurs and X_2 occurs…., and X_n occurs,

all together. To classify an item, the task is to calculate the $P\{y_i|X_1, X_2, \dots, X_n\}$ for each class y_i, and then choose the class with the largest probability as the best fit for the item.

The Naïve Bayes method solves the problem of the extreme computational task of calculating the joint probabilities for the information which includes evaluation of many combinations and the need for extremely large volumes of data to cover them all, by making the simplifying assumption that the predictors X_1, X_2 ,…., X_n are independent of each other. Under this assumption, we can use the following expression:

$$P\{X_1, X_2, \dots, X_n|y_i\} = P\{X_1|y_i\} * P\{X_2|y_i\} * \dots . P\{X_n|y_i\}$$

This is because when events are independent, the joint probability of multiple events occurring is simply the product of the probabilities of the individual events. The method can now again use frequency counts in the training data, this time for the occurrences of the predictors within each class. For example, $P\{X_i|y_i\}$ is estimated by counting how many times X_1 occurs in class y_i, divided by the total number of items in the class.

The result is that in the Naïve Bayes we use the expression below, where all the quantities on the right-hand side are obtained directly from frequency counts in the training data.

$$P\{y_i|X_1, X_2, \dots, X_n\}$$

$$= \frac{P\{X_1|y_i\} * P\{X_2|y_i\} * \dots . P\{X_n|y_i\}P\{y_i\}}{(P\{X_1|y_1\} * P\{X_2|y_1\} *** P\{X_n|1\})P\{y_1\} + .. + (P\{X_1|y_m\} * P\{X_2|y_m\} * .. P\{X_n|y_m\})P\{y_m\}}$$

After these probabilities are obtained from the training data, they can be applied to classify new data that was never used in training. Thus, the Naïve Bayes is an example of supervised learning in practice. Overall, Naïve Bayes is a simple, computationally efficient, and usually does an accurate job of classification [8]. The Naive Bayes method often outperforms

more sophisticated classification methods. [9] Following there is a simple example showing how Naïve Bayes works.

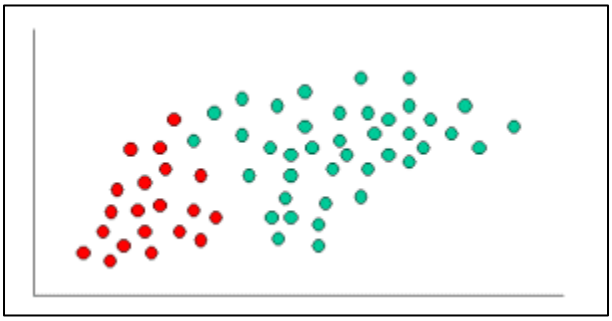


Figure 1. Objects classified as either green or red

Figure 1 demonstrates the concept of Naïve Bayes classification. As indicated, the objects can be classified as either green or red. The task is to classify new cases as they arrive, i.e., to decide the class label to which they belong, based on the current objects.

Because there are twice as many green objects as there are red ones, it is reasonable to believe that a new case is twice as likely to have membership in the green group rather than the red one. With Bayesian analysis, this is called the prior probability. Prior probabilities are based on previous experience, in this case, the percentage of green and red objects, and are often used to predict outcomes before they happen. There, we can write:

$$Prior\ probability\ for\ green \propto \frac{Number\ of\ green\ object}{Total\ number\ of\ objects}$$

$$Prior\ probability\ for\ red \propto \frac{Number\ of\ red\ object}{Total\ number\ of\ objects}$$

Because there is a total of 60 objects, 40 of which are green and 20 of which are red, our prior probabilities for the class membership are as follows:

$$Prior\ probability\ for\ green \propto \frac{40}{60}$$

$$Prior\ probability\ for\ red \propto \frac{20}{60}$$

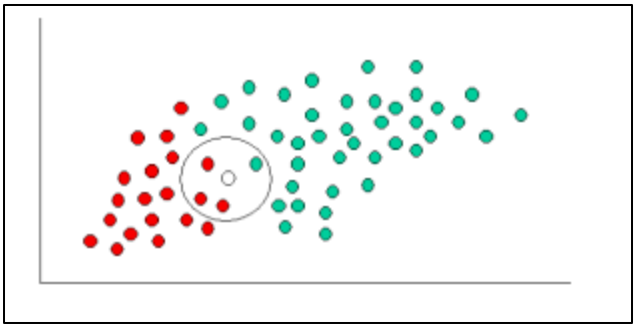


Figure 2. Classify new object (white circle)

After formulating the prior probability, we can classify a new object (white circle) in Figure 2. Because the objects are well clustered, it is reasonable to assume that the more green (or red) objects that are in the area of X, the more likely it is that the new cases belong to that particular color. To measure this likelihood, we draw a circle around X; this circle encompasses a number of points, irrespective of their class labels. Then, we calculate the number of points in the circle that belongs to each class label. From this result, we calculate the likelihood:

$$Likelihood\ of\ X\ given\ green \propto \frac{Number\ of\ green\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ green\ cases}$$

$$Likelihood\ of\ X\ given\ red \propto \frac{Number\ of\ red\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ red\ cases}$$

From Figure 2, it is clear that the Likelihood of X given green is smaller than the Likelihood of X given red because the circle has 1 green object and 3 red ones. Therefore,

$$Probability\ of\ X\ given\ green \propto \frac{1}{40}$$

$$Probability\ of\ X\ given\ red \propto \frac{3}{20}$$

Although the prior probabilities indicate that X may belong to green, the likelihood indicates otherwise: that the class membership of X is red. In the Bayesian analysis, the final classification is produced by combining both information sources, i.e., the prior probability and the likelihood, to form a posterior probability using the Bayes rule.

$$Posterior\ probability\ of\ X\ given\ green$$

$$\propto Prior\ probability\ of\ green \times Likelyhood\ of\ X\ given\ green$$

$$= \frac{4}{6} \times \frac{1}{40} = \frac{1}{60}$$

$$Posterior\ probability\ of\ X\ given\ red$$

$$\propto Prior\ probability\ of\ red \times Likelihood\ of\ X\ given\ red$$

$$= \frac{2}{6} \times \frac{3}{20} = \frac{1}{20}$$

Finally, we classify X as red because its class membership achieves the largest posterior probability [10] .

The idea behind naïve Bayes classifier is trying to compute the script's probability of being positive or negative by mining the text. As a supervised learning method, the Naïve Bayes classifier often performs very well in practice. Naïve Bayes is also very simple, so if we are looking for a straightforward and good performance method, then the Naïve Bayes classifier is a good option. In the Sentiment package of R, the Naïve Bayes classifier is used, and the Naïve Bayes classification is adopted. Chapter 3 shows more detail about the Sentiment package.

2.3. Support Vector Machine

The Support Vector Machine (SVM) is a well-known, supervised machine-learning algorithm to perform text classification [11] [12]. The main point of a support vector machine is to find a linear separator in the search space that can best separate the different classes. Support vectors are the data points that lie closest to the decision surface and have a direct bearing on the optimum location of the decision surface. SVMs maximize the margin around the separating hyperplane. The decision function is fully specified by a subset of training samples, the support vectors. Text classification method is available by SVM [13].

As Figure 3 shows, there are two classes, A and B. The three hyperplanes between them, I, II and III, separate them into two classes. We will choose the hyperplane which has the largest normal distance for any data points as the best separator, hyperplane I in Figure 3 [4]

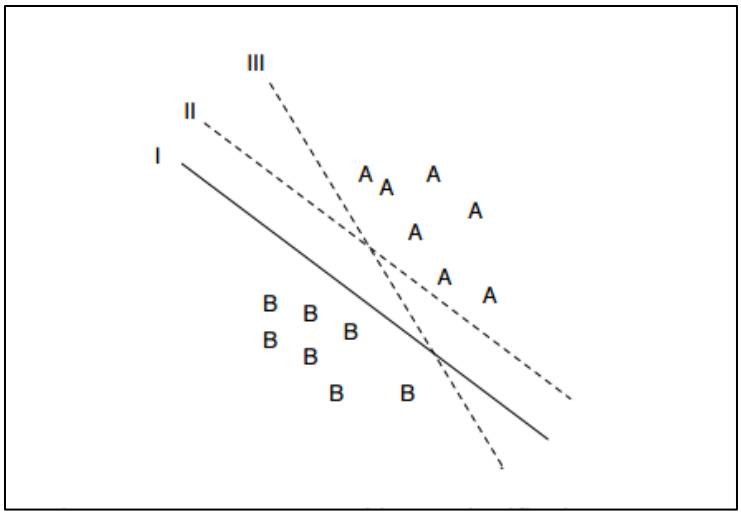


Figure 3. Support Vector Machine (SVM) for classification

Support Vector Machines (SVM) have been shown to be highly effective with traditional text categorization. They are large-margin, rather than probabilistic classifiers, in contrast to the Naive Bayes method. In my experiments, I am working with Twitter feeds which are text, and SVMs work well for text classification [14].

2.4. Related Work

Overall, text classification using machine learning is a well-studied field [15]. Pang and Lee researched the effects of various machine-learning techniques [Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machines (SVM)] in the specific domain of movie reviews [16]. Researchers have also worked on detecting sentiments in the text. Turney presents a simple algorithm, called semantic orientation, to detect sentiments [6]. Pang and Lee present a hierarchical scheme where the text is first classified as containing a sentiment and then classified as positive or negative [7].

A considerable amount of sentiment analysis research has been done with Twitter. Apoorv A., Boyi X., et al. researched the use of a tree kernel to remove the need for tedious feature engineering [17]. Bo Yuan researched the methodologies applied for a sentiment classification of Twitter data: lexicon-based, rule-based, and machine learning-based methods [14]. Agarwal and Sabharwal [18] extracted and analyzed a single tweet and its followers with sentiment analysis. Another significant effort for sentiment classification of Twitter data was done by Barbosa and Feng. They used polarity predictions from three websites as noisy labels to train a model; they used 1,000 manually labeled tweets for tuning and another 1,000 manually labeled tweets for testing [19]. They did not mention how they collected their test data. They proposed using the tweets' syntax features, such as retweets, hashtags, links, punctuation, and exclamation marks, in conjunction with features such as the words' prior polarity.

3. SYSTEM DESIGN

Even though some studies have been done to analyze Twitter, I do not see any work to evaluate positive and negative views by relying on Twitter analyses for market research and comparison. This chapter presents more details about what I have done with the Twitter research. The chapter describes the process of capturing Twitter feeds, parsing a dictionary, identifying the polarity of Twitter feeds, tallying positive and negative words in a tweet, and conducting further analysis.

3.1. Capturing Twitter Feeds into R Studio

In order to use a streaming API to capture tweets, I needed to take the following steps.

3.1.1. Getting Twitter API Keys

First, we need to create a Twitter account. Then, we should visit the Twitter application's management website (https://apps.twitter.com/) to create a new app and to complete the form (Figure 4).

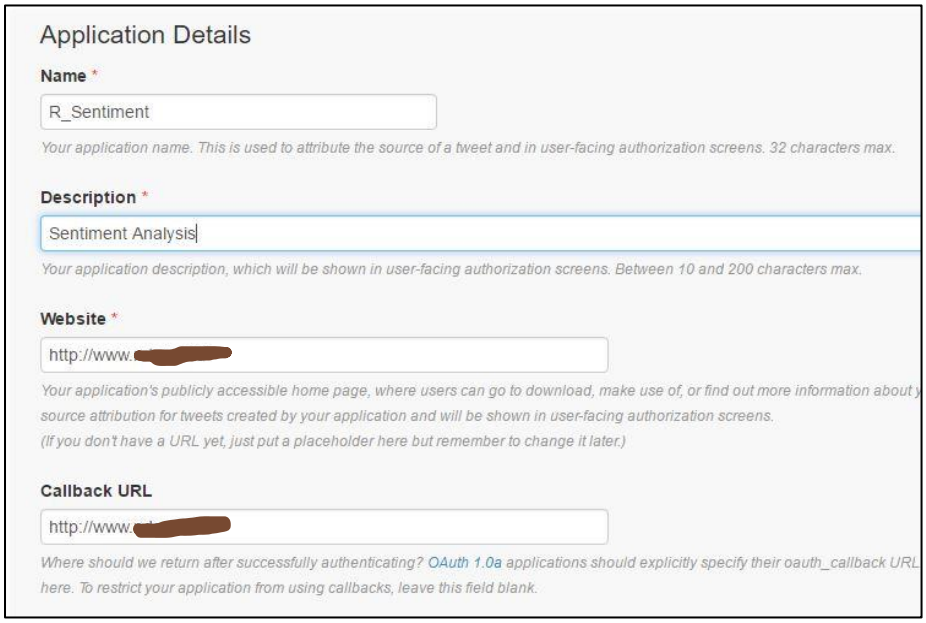


Figure 4. Create a new app

After the Twitter app has been created, we can get four keys: “API key,” “API secret,” “Access token,” and “Access token secret.” Figure 5 shows the four keys. Those four keys contain the user’s credentials to access the Twitter API.

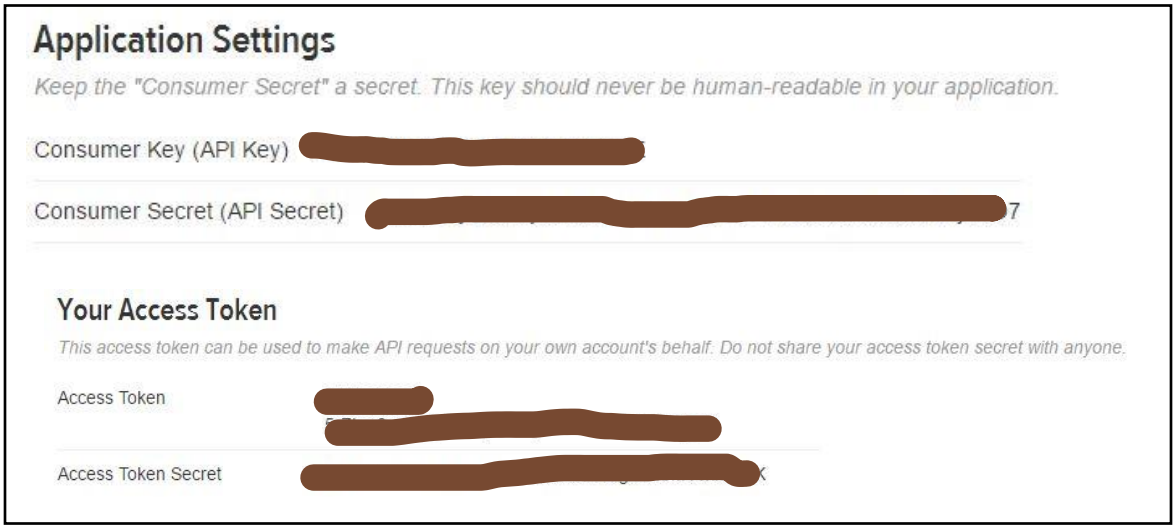


Figure 5. Get API keys

3.1.2. Start to Extract Tweets in R

After successfully getting the keys, we need to start writing R scripts in R Studio. We need to install the TwitteR package in R Studio. We need to write the API key, API secret, Access token, and Access token secret information; then, we need to write the codes to relate the Twitter feeds to the keyword we selected.

3.2. Data-Vector Representation and Cleaning/Formatting

After getting the Twitter feeds related to the selected keyword, we need to do the text’s vector representation, and then clean or format the data so that we can run the sentiment analysis in R studio. We need to use “sapply” in R which applies a function to elements in the list of tweets and then returns the results in a matrix. Then, we need to write the code to remove the retweets, ID names, punctuation, numbers, HTML links, extra spaces, etc. so that we obtain clean data with which to work. Figure 6 shows the flow for clearing and formatting the tweet.

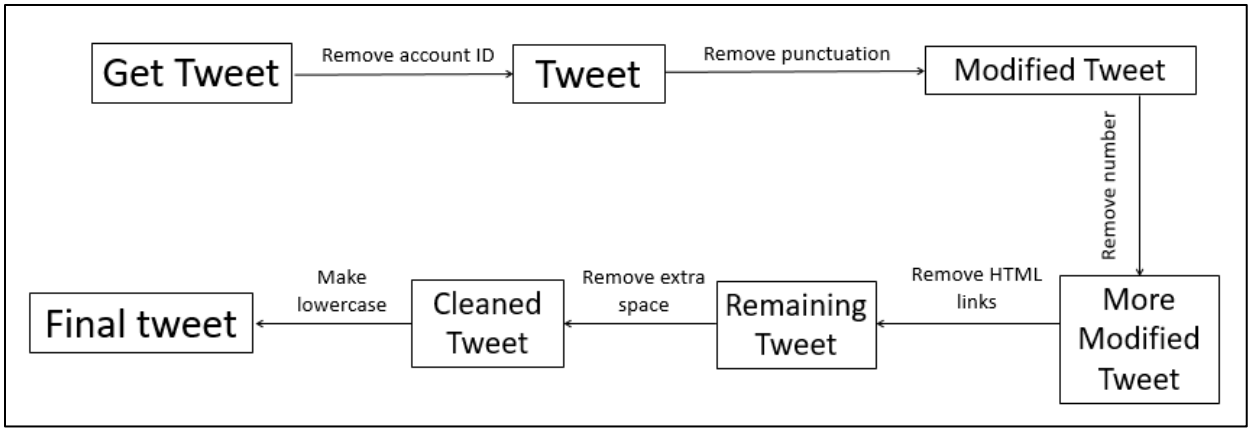


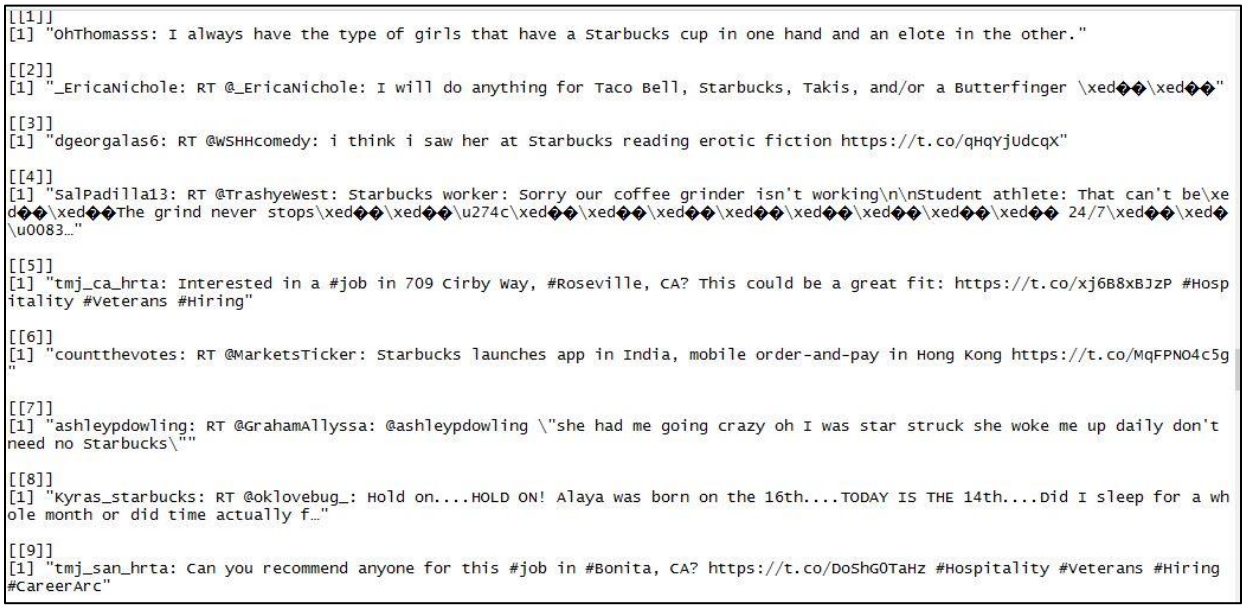
Figure 6. Flow for cleaning/formatting tweets

There is the screenshot of the code I have used for cleaning/formatting the data in figure 7. I have used these codes in R-studio to obtain a clean set of data on which I can run the sentiment analysis.

```
# remove at people
some_txt = gsub('@\\w+', '', some_txt)
# remove punctuation
some_txt = gsub('[[:punct:]]', '', some_txt)
# remove numbers
some_txt = gsub('[[:digit:]]', '', some_txt)
# remove html links
some_txt = gsub('http\\w+', '', some_txt)
# remove unnecessary spaces
some_txt = gsub('[ \t]{2,}', '', some_txt)
some_txt = gsub('^\\s+|\\s+$', '', some_txt)

# define 'tolower error handling' function
try.error = function(x)
{
  # create missing value
  y = NA
  # tryCatch error
  try_error = tryCatch(tolower(x), error=function(e) e)
  # if not an error
  if (!inherits(try_error, 'error'))
    y = tolower(x)
  # result
  return(y)
}
# lower case using try.error with sapply
some_txt = sapply(some_txt, try.error)
```

Figure 7. Sample codes for cleaning/formatting data

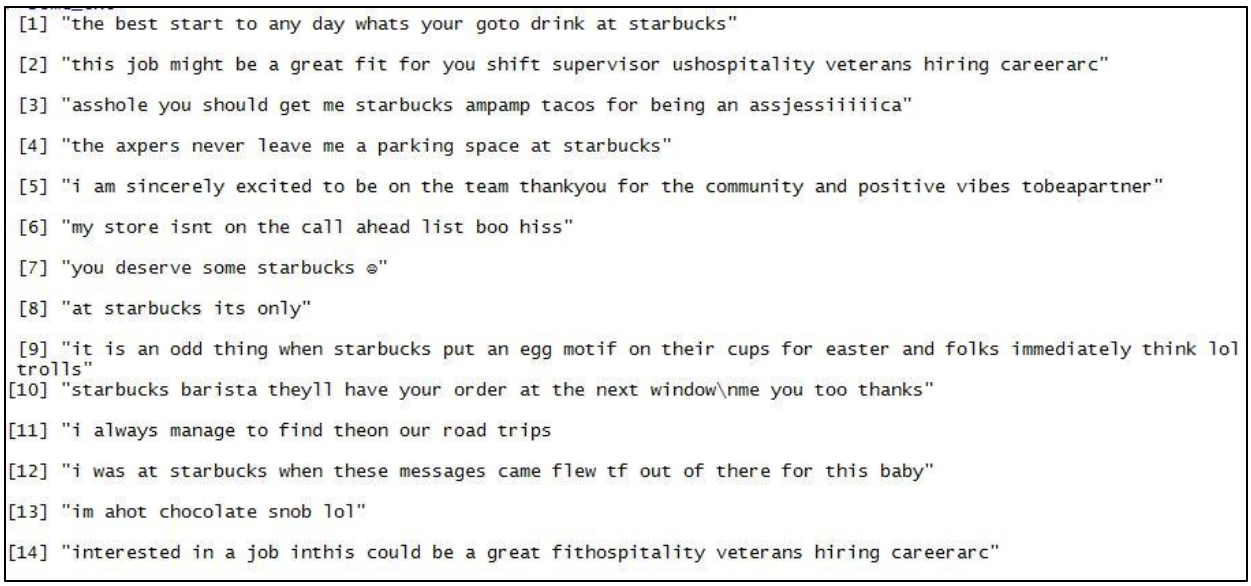
There is an example for the keyword “Starbucks.” Figures 8 and 9 show the parsed tweets without and with the cleaning process, respectively.



```
[[1]]
[1] "OhThomasss: I always have the type of girls that have a Starbucks cup in one hand and an elote in the other."

[[2]]
[1] "_EricaNichole: RT @_EricaNichole: I will do anything for Taco Bell, Starbucks, Takis, and/or a Butterfinger \xed\xed"

[[3]]
[1] "dgeorgalas6: RT @WSHHcomedy: i think i saw her at Starbucks reading erotic fiction https://t.co/qHqYjUdcqX"

[[4]]
[1] "SalPadilla13: RT @TrashyeWest: Starbucks worker: Sorry our coffee grinder isn't working\n\nStudent athlete: That can't be\xe
d\xed\xedThe grind never stops\xed\xed\u274c\xed\xed\xed\xed\xed\xed\xed\xed\xed 24/7\xed\xed\xed
\u0083…"

[[5]]
[1] "tmj_ca_hrta: Interested in a #job in 709 Cirby Way, #Roseville, CA? This could be a great fit: https://t.co/xj6B8xBJzP #Hosp
itality #Veterans #Hiring"

[[6]]
[1] "countthevotes: RT @MarketsTicker: Starbucks launches app in India, mobile order-and-pay in Hong Kong https://t.co/MqFPNO4c5g
"

[[7]]
[1] "ashleypdowling: RT @GrahamAllyssa: @ashleypdowling \"she had me going crazy oh I was star struck she woke me up daily don't
need no Starbucks\""

[[8]]
[1] "Kyras_starbucks: RT @oklovebug_: Hold on....HOLD ON! Alaya was born on the 16th....TODAY IS THE 14th....Did I sleep for a wh
ole month or did time actually f…"

[[9]]
[1] "tmj_san_hrta: Can you recommend anyone for this #job in #Bonita, CA? https://t.co/DoShG0TaHz #Hospitality #Veterans #Hiring
#CareerArc"
```

Figure 8. Tweets without the cleaning/formatting process

From Figure 8, we can see there are retweets, ID names, punctuation, numbers, HTML links, extra spaces, etc. In Figure 9, we can only see the texts, without any retweets, punctuation, numbers, HTML links, and extra spaces.



```
[1] "the best start to any day whats your goto drink at starbucks"

[2] "this job might be a great fit for you shift supervisor ushospitality veterans hiring careerarc"

[3] "asshole you should get me starbucks ampamp tacos for being an assjessiiiiica"

[4] "the axpers never leave me a parking space at starbucks"

[5] "i am sincerely excited to be on the team thankyou for the community and positive vibes tobeapartner"

[6] "my store isnt on the call ahead list boo hiss"

[7] "you deserve some starbucks ☕"

[8] "at starbucks its only"

[9] "it is an odd thing when starbucks put an egg motif on their cups for easter and folks immediately think lol
trolls"
[10] "starbucks barista theyll have your order at the next window\nme you too thanks"

[11] "i always manage to find theon our road trips

[12] "i was at starbucks when these messages came flew tf out of there for this baby"

[13] "im ahot chocolate snob lol"

[14] "interested in a job inthis could be a great fithospitality veterans hiring careerarc"
```

Figure 9. Tweets with the cleaning/formatting process

3.3. Sentiment Analysis

In this research, I did a sentiment analysis using the Naïve Bayes and Support Vector Machine (SVM) classifiers. Both methods are explained in the following sections.

3.3.1. Naïve Bayes Classifier

To use the Naïve Bayes classifier in R, I needed to install Rstem and the Sentiment package from the R-CRAN repository archive. The Sentiment package uses a classifier based on the Naive Bayes algorithm. It was built to use a trained dataset of emotion words (more than 6,500 words) [20]. Table 1 shows the example of positive and negative words from the subjectivity lexicon. In R, I used a function, classify_polarity(), provided by the Sentiment package, to classify the tweets into two classes, pos (positive sentiment) and neg (negative sentiment), for the Naïve Bayes classification.

Table 1. Example of positive and negative words from subjectivity lexicon

Words	**Subjectivity**
awkward	Negative
awesome	Positive
terrible	Negative
favorite	Positive
hate	Negative
terrific	Positive

3.3.2. SVM Classifier

For the SVM classification, I utilized the same dataset (tweets) which was used for the Naïve Bayes analysis. I used a dictionary with more than 6,500 polarity words to determine each tweet's polarity (positive/ negative) by matching the words with each tweet. When there was

more than one word that was matched in a tweet, the polarity (positive/ negative) was defined by the highest number of the tweet's polarity words. For example, if there were two positive words and one negative word in a tweet, then that tweet was defined as positive. After determining the polarity by using the R script, I went through all the data manually in order to check the polarity. From this dataset, I used 70% of the data as training data and tested 30% of the data with the SVM classifier. In R, I needed to install the e1071 and RTextTools packages in order to conduct the SVM analysis.

3.4. Experiments with the Model

For this project, the model could be used as a tool for a business' market analysis in order to obtain people's perception about the business through Twitter feeds. I conducted the experiment with two very well-known supermarkets, Walmart and Target, just before Christmas, the time when those stores have big sales. I conducted two separate experiments with two keywords: "Walmart" and "Target supermarket."

3.5. Precision and Recall

Precision and recall are the basic measures used for evaluating strategies, typically used in document retrieval. In this project, I calculated the precision and recall for both algorithms, Naïve Bayes and SVM.

Precision: Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records that were retrieved. Precision is usually expressed as a percentage.

Recall: Recall is the ratio of the number of relevant records that were retrieved to the total number of relevant records in the database. It is usually expressed as a percentage [21].

Calculating precision and recall: Let us suppose that there are 100 positive cases among 10,000 cases. We want to predict which ones are positive, and we pick 200 of them to have a better chance of catching many of the 100 positive cases. We record the IDs for our predictions, and when we obtain the actual results, we sum up how many times we were right or wrong. There are four ways to be right or wrong:

TN/True Negative: The case was negative and predicted negative.

TP/True Positive: The case was positive and predicted positive.

FN/False Negative: The case was positive but predicted negative.

FP/False Positive: The case was negative but predicted positive.

Now, we can count how many of the 10,000 cases fall into each category. Let us assume that TP, TN, FN, and FP were found as shown in Table 2.

Table 2. Example of TP, TN, FN, and FP

	Predicted Negative	Predicted Positive
Negative Cases	TN: 9,760	FP: 140
Positive Cases	FN: 40	TP: 60

The equations for calculating precision and recall are as follows:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Now, if we want to find the precision, the percentage of positive predictions that were correct, the answer is as follows: Precision = 60/ (140+60) = (60/200) ×100 = 30%. When we want to find the recall, the percentage of positive cases that we could catch, the answer is as follows: Recall = 60/ (60+40) = (60/100) ×100 = 60% [22].

4. EXPERIMENT AND RESULT

In this chapter, the experiments and the results are discussed. The dataset that I utilized, the experiments, the results, and a comparison of the results are discussed, in detail, in this chapter.

4.1. Dataset

I conducted the experiment with two very well-known supermarkets, Walmart and Target, using the words "Walmart" and "Target supermarket" on December 23rd and December 24th of 2016. I searched for 15,000 tweets for each experiment, and after removing the retweets, I had 11,695 tweets for the Walmart experiment and 10,560 tweets for Target supermarket experiment. I have captured the streaming tweet, so the tweets were contingent in time.

4.2. Experiments

4.2.1. Estimating Sentiment with the Naïve Bayes Method

For the Naïve Bayes analysis, I installed the Rstem and Sentiment packages from the R-CRAN repository archive in R studio. The Sentiment package was built to use a trained dataset of emotion words (more than 6,500 words). In R, I used the function classify_polarity(), from the sentiment package, to categorize the tweets into two classes, pos (positive sentiment) and neg (negative sentiment), for the Naïve Bayes classification for both the Walmart and Target experiments. For simplicity, I avoided the neutral tweets. After classification, all the tweets were saved in a .csv file, and I counted the total positive and negative tweets using the length () function.

4.2.1.1. Result from the experiment using the Naïve Bayes algorithm and comparison

In this project, I searched for 15,000 tweets for each experiment, and after removing the retweets, I had 11,695 tweets for the Walmart experiment and 10,560 tweets for the Target supermarket experiment. The Sentiment package was built to use a trained dataset of emotion words (more than 6,500 words). Using the function and the trained dataset from the sentiment package, I obtained the results (positive and negative) for both the Walmart and Target experiments. Table 3 shows the detailed results.

Table 3. Results from using Naïve Bayes for the Walmart and Target experiments

Experiments	Total	Positive	Negative	Positive (%)	Negative (%)
Walmart	11695	8009	3686	68.48%	31.52%
Target	10560	7477	3083	70.80%	29.20%

From the experiments using the Naïve Bayes algorithm, I obtained the results for Walmart and Target. Figure 10 shows the results in percentiles because the data sizes were different.

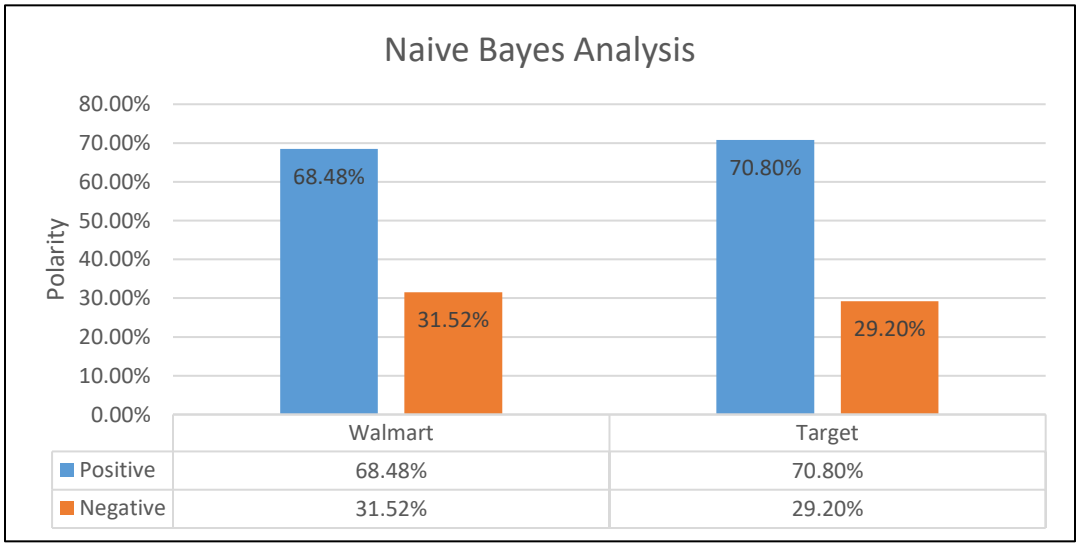


Figure 10. Naïve Bayes analysis for Walmart and Target

From Figure 10, we can see that, for the Walmart experiment, 68.48% of the tweets were positive about Walmart while 31.52% of the tweets were negative. For Target supermarket, 70.80% of the tweets were positive, and 29.20% of the tweets were negative. According to this experiment's results, there was a slightly more positive opinion about Target than Walmart.

4.2.2. Estimating Sentiment with Support Vector Machine

For the SVM classification, I utilized the same dataset (tweets) which was used for the Naïve Bayes analysis for the Walmart and Target experiments. I used a dictionary with more than 6,500 polarity words from the R-CRAN archive to determine each tweet's polarity (positive/negative) by matching the words with each tweet. In R, I installed the e1071 and RTextTools packages to conduct the SVM analysis. From this dataset, I used 70% of the data as training data and ran the tests, utilizing the SVM classifier, on 30% of the data for the Walmart and Target experiments. After running the tests, I saved the document summary in .csv file and calculated the positive and negative tweets for each Walmart and Target experiment.

4.2.2.1. Result from the experiment using the Support Vector Machine Algorithm

In this project, I searched for 15,000 tweets for each experiment, and after removing the retweets, I had 11,695 tweets for the Walmart experiment and 10,560 tweets for the Target supermarket experiment. I used 70% of the data as training data and ran the tests, utilizing the SVM classifier, on 30% of the data for both experiments. For the Walmart experiment, I had 3,509 (30% of 11,695 tweets) tweets, and for the Target experiment, I had 3,168 (30% of 10,560 tweets) tweets for testing. Using functions from the e1071 and RTextTools packages, I obtained the results (positive and negative) for both the Walmart and Target experiments. Table 4 shows the detailed results.

Table 4. Results from the experiment using SVM for the Walmart and Target Experiments

Experiment	Total	Positive	Negative	Positive (%)	Negative (%)
Walmart	3509	2357	1152	67.17%	32.83%
Target	3168	2255	913	71.18%	28.82%

Results from the experiment with the Walmart and Target data using the SVM are illustrated in Figure 11.

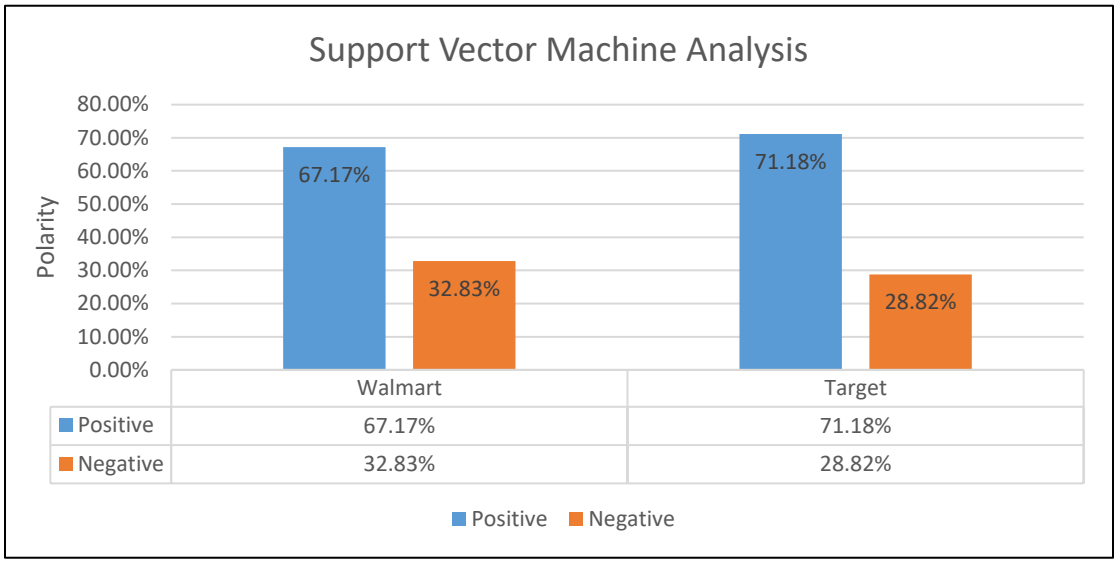


Figure 11. SVM analysis for Walmart and Target

From Figure 11, we can see that, for the Walmart experiment, 67.17% of the tweets were positive about Walmart while 32.83% of the tweets were negative. For Target supermarket, 71.18% of the tweets were positive, and 28.82% of the tweets were negative. From these results, we can say that there was a more positive opinion about Target than Walmart.

4.3. Overall Comparison of the Results from the Two Algorithms

After doing the experiments with the Walmart and Target supermarket data, we found the sentiment analysis' results for both the Naïve Bayes and SVM algorithms. For the Walmart experiment, 68.48% of the tweets were positive while 31.52% of the tweets were negative when using the Naïve Bayes algorithm. At the same time, 67.17% of the tweets were positive about Walmart while 32.83% of the tweets were negative when using the SVM algorithm. For both algorithms, we found similar results for the Walmart experiment. Figure 12 shows the results of the Walmart experiment.

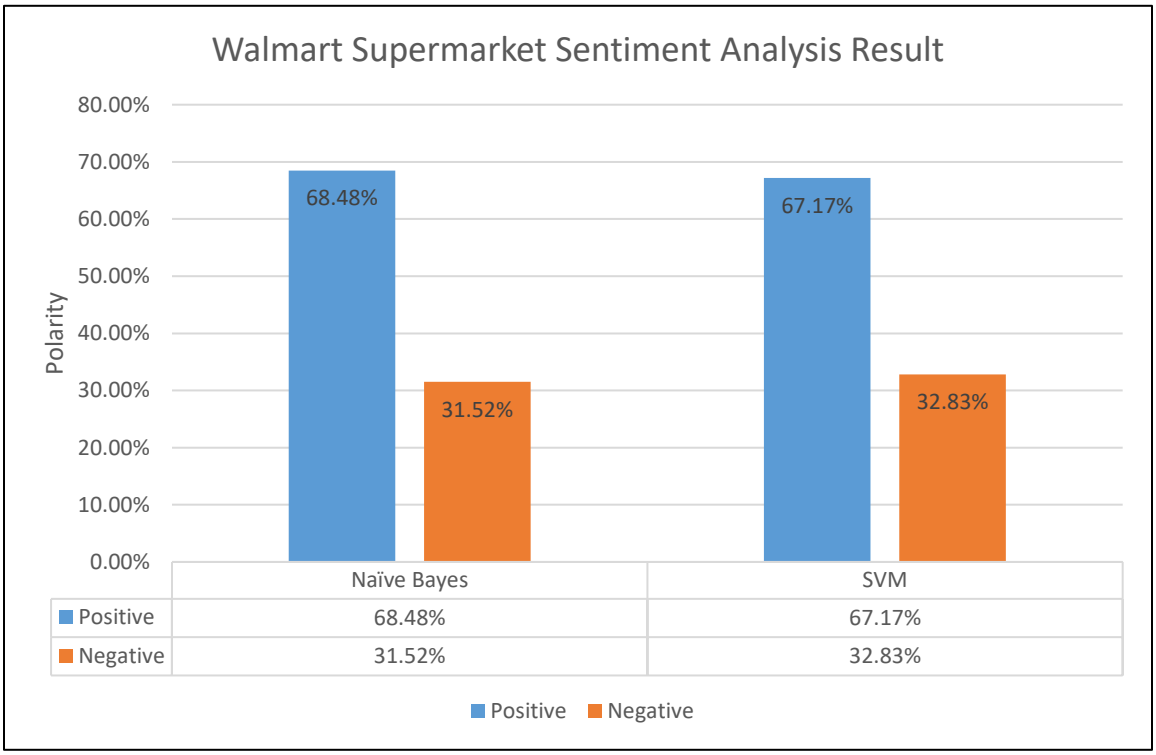


Figure 12. Walmart supermarket sentiment-analysis results for the Naïve Bayes and SVM methods.

For the Target supermarket experiment, 70.80% of the tweets were positive while 29.20% of the tweets were negative when using the Naïve Bayes algorithm. At the same time, 71.18% of the tweets about the Target supermarket were positive while 28.82% of the tweets were negative when using the SVM algorithm. For both algorithms, we found that the Naïve Bayes and SVM methods have almost the same result for the Target supermarket experiment. Figure 13 shows the results.

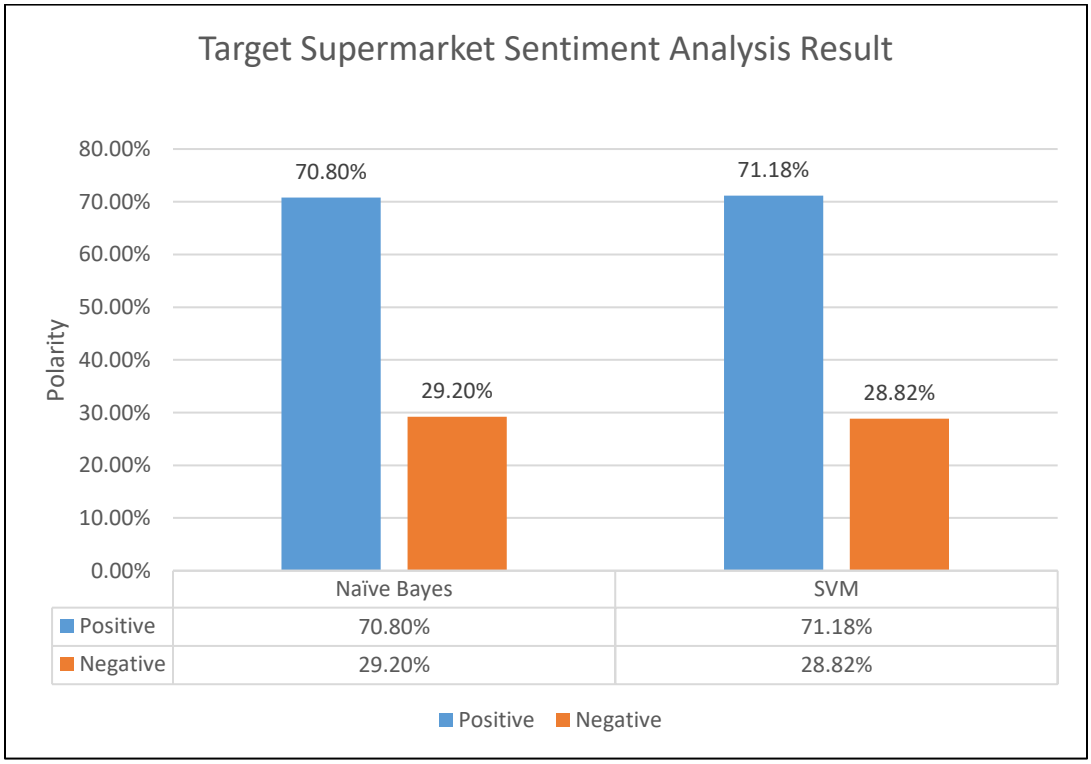


Figure 13. Target supermarket sentiment-analysis results for the Naïve Bayes and SVM methods

4.4. Accuracy of the Algorithms

4.4.1. Accuracy of the Naïve Bayes Algorithm: Precision and Recall

From the experiment with Walmart and Target supermarket data using the Naïve Bayes algorithm, we tested the Naïve Bayes method's precision and recall to observe the accuracy. For the Walmart data's sentiment analysis, the precision was 93.2% while the recall was 80%, and for the Target supermarket analysis, the precision was 85% while the recall was 81%. Therefore,

we can say that, for the Walmart experiment, the percentage of correct positive predictions was 93.2%, and the percentage of positive cases that we could catch was 80%. On the other hand, for the Target supermarket experiment, the percentage of correct positive predictions was 85%, and the percentage of positive cases that we could catch was 81% when using the Naïve Bayes algorithm. Figure 14 shows the result.

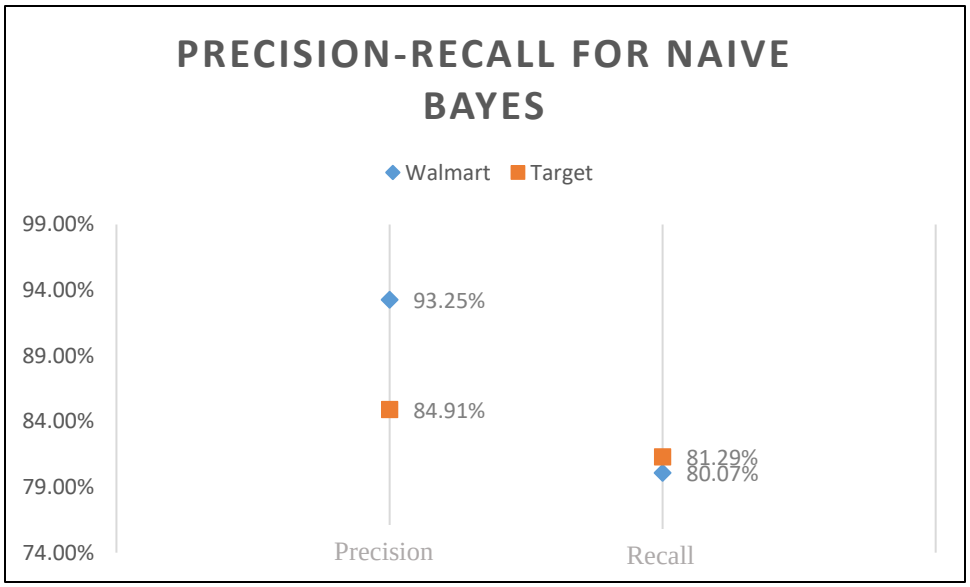


Figure 14. Precision-Recall for the Naive Bayes algorithm

4.4.2. Accuracy of the Support Vector Machine (SVM) Algorithm: Precision and Recall

From the experiment with the Walmart and Target supermarket data using the SVM algorithm, we tested the SVM's precision and recall in order to ascertain the accuracy. For the Walmart data's sentiment analysis, the precision was 81.5% while the recall was 76.5%, and for the Target supermarket analysis, the precision was 81% while the recall was 79%. Therefore, for the Walmart experiment, the percentage of correct positive predictions was 81.5%, and the percentage of positive cases that we could catch was 76.5%. On the other hand, for the Target supermarket experiment, the percentage of correct positive predictions was 81% while the percentage of positive cases that we could catch was 79% when using the SVM algorithm. Figure 15 shows the results on the next page.

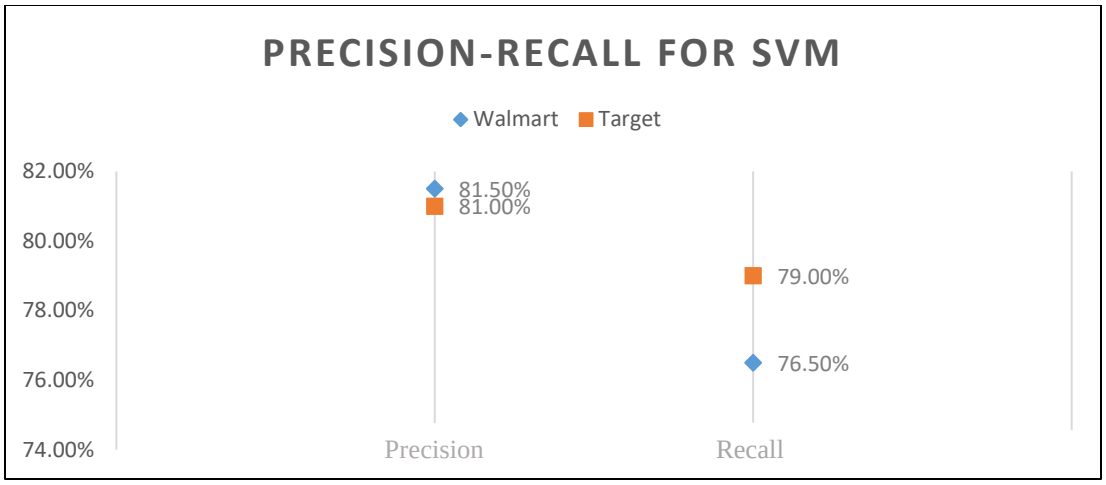


Figure 15. Precision and Recall for SVM

4.4.3. 10-fold Cross-Validation

Cross-validation is a technique that is used to evaluate predictive models by partitioning the original sample into a training set to train the model and a test set to evaluate the model [23]. 10-fold cross-validation breaks the data into 10 sets of size n/10. It trains on 9 datasets and tests on 1, repeating the process 10 times and taking the mean accuracy. I did the 10-fold cross validation for the SVM analysis with the Walmart and Target data; the results are shown in figure 16 and figure 17.

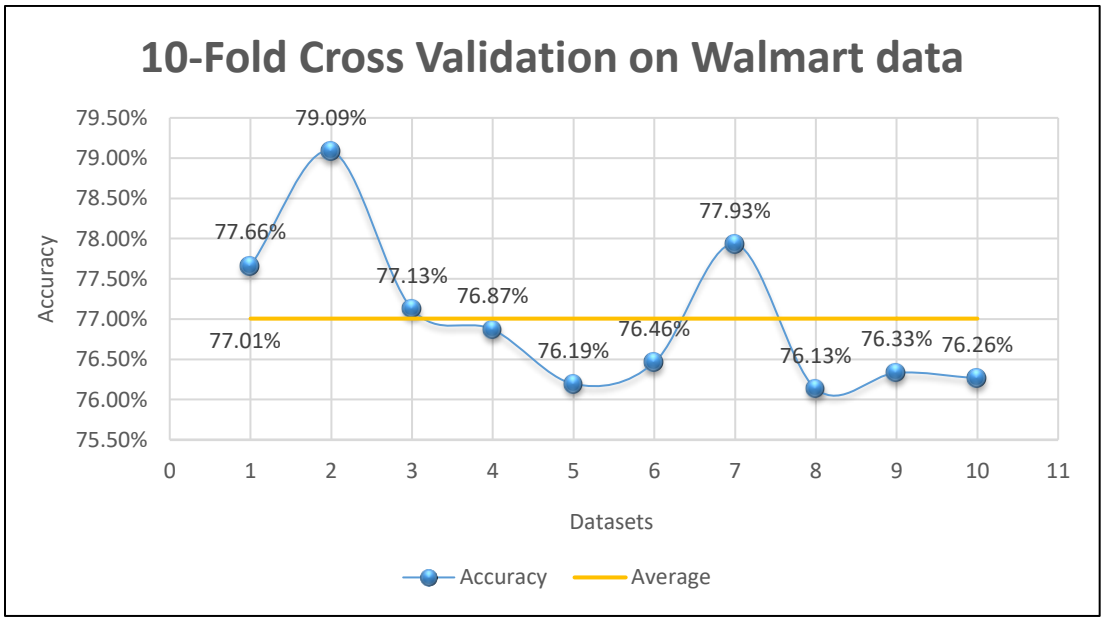


Figure 16. 10-fold cross validation with the Walmart data.

Figure 16 shows the 10-fold cross validation for the Walmart experiment. The figure shows the mean accuracy of 10 times in the cross-validation process, which breaks the Walmart data into 10 sets of size n/10. It trains on 9 datasets and tests on 1 set, with 10 repeats. Here, we can see a mean accuracy of 77.66%, 79.09%, 77.13%, 76.87%, 76.19%, 76.46%, 77.93%, 76.13%, 76.33%, and 76.26%. The average of these ten accuracies is 77.01%. The cross validation shows the predictive performance of the SVM model. From figure 15, we can see that the accuracy of 2nd set of the data is highest and accuracy of 8th set of date is the lowest. There are very little differences in accuracy between all the sets of data but 1st, 2nd, 3rd and 7th set has accuracy more than the average.

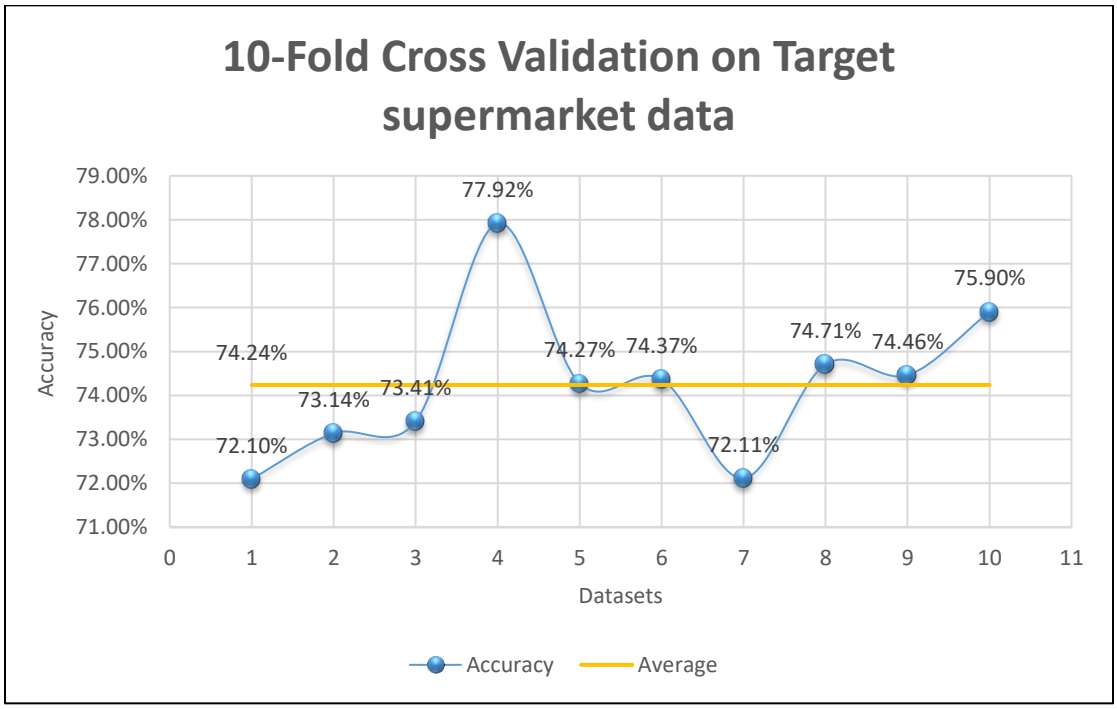


Figure 17. 10-fold cross validation for the Target supermarket data.

Figure 17 shows the 10-fold cross validation for the Target supermarket experiment. The figure shows the mean accuracy for 10 times with the cross-validation process, which breaks the Target supermarket data into 10 sets of size n/10. It trains on 9 datasets and tests 1 set, with 10 repeats. Here, we can see a mean accuracy of 72.10%, 73.14%, 73.41%, 77.92%, 74.27%,

74.37%, 72.11%, 74.71%, 74.46%, and 75.90%. The average of these ten accuracies is 74.24%. The cross validation shows the predictive performance of the SVM model. From figure 16, we can see that the accuracy of 4th set of the data is highest (77.92%) and accuracy of the 1st set of date is the lowest (72.10%). There are very little differences in accuracy between all the sets of data but 4th,5th, 6th, 8th, 9th, and 10th set has accuracy more than the average.

4.4.4. Coefficient of Variation

To obtain the degree of variation of the accuracies from 10-fold cross validation for the datasets of Walmart and Target supermarket data, I have calculated the coefficient of variation. The coefficient of variation is a measure of spread that describes the amount of variability relative to the mean.

$$Coefficient\ of\ Variation\ (Cv) = \frac{Standard\ Deviation\ (\sigma)\times 100}{Mean\ (\mu)}$$

Coefficient of variation for cross validation on Walmart data,

$$Coefficient\ of\ Variation\ (Cv) = \frac{0.00964304}{0.7701}\times 100 = 1.25\%$$

Coefficient of variation for cross validation on Target supermarket data,

$$Coefficient\ of\ Variation\ (Cv) = \frac{0.017517311}{0.7424}\times 100 = 2.36\%$$

From the results, we can say that there is insignificant variation in the accuracies for all the datasets in 10-fold cross validation, for both Walmart and Target supermarket data. So, we can say that the model worked consistently for all the datasets.

5. CONCLUSION

For this paper, my goal was to understand people's opinions as market analysis, concerning different businesses. Therefore, I designed the experiments and followed the steps. First, I obtained the API key, API secret, Access token, and Access token secret. Then, I connected to the Twitter Streaming API and started to capture the Twitter feeds for Walmart. The next step was to remove retweets, punctuation, numbers, HTML links, extra spaces, etc. from the captured data so that I had clean data with which to work. Then, I ran the analysis with both the Naïve Bayes and SVM algorithms. I repeated all the steps for the Target supermarket experiment.

Finally, the experimental result illustrated the positive and negative attitudes, in percentages, towards those stores. There was a slightly more positive attitude towards the Target supermarket compared to Walmart when using both algorithms. The algorithms' precision and recall showed the accuracy for each experiment. The Naïve Bayes algorithm was precise and sensitive for both the Walmart and Target supermarket experiments. The SVM algorithm was good at precision and recall for both the Walmart and Target supermarket experiments. In this project, the Naïve Bayes algorithm won by a small percentage. Comparison between Walmart and Target supermarket shows that people have more positive opinion for Target supermarket than Walmart. Target is a winner here.

6. FUTURE WORK

This chapter describes how we can extend the study in the future.

6.1. Sentiment Analysis for Big Data

The ability to exploit public sentiment in social media is increasingly considered to be an important tool for market understanding, consumer segmentation, and stock-price prediction for strategic marketing's planning and guiding. This evolution of technology adoption is energized by healthy growth in the big-data framework; the growth caused applications based on Sentiment Analysis of big data to become common for businesses. However, few works have studied the gaps for the Sentiment-Analysis application in big data. Although Sentiment Analysis is the main agenda item with big data, no known work has discussed whether the Sentiment-Analysis approaches are suitable for big data's infrastructure [24].

6.2. Twitter Analysis on Other Aspects

I did experiments with Walmart and Target supermarket's Twitter data. Many commercial or advertisement companies can conduct a Twitter Sentiment Analysis to understand people's purchasing attitudes about their products. The companies can do the market research, can create a new product line, and can improve their products' quality. A social network makes the world smaller, so it is a very good source to discover what is happening in the different world sectors and to learn people's opinions about those things. For example, investors can use Twitter to discern investment confidence for different investments.

7. REFERENCES

[1] E. McQuarrie, The market research toolbox: a concise guide for beginners, 2nd ed., SAGE, 2005.

[2] ICC/ESOMAR, "International Code on Market and Social Research," in *International Chamber of Commerce (ICC)*, Amsterdam, 2008.

[3] J. Bollen, M. Huina and Z. Xiaojun, "Twitter Mood Predicts the Stock Market," *Computational Science 2.1,* pp. 1-8, 2011.

[4] Z. Li, "Naive Bayes Algorithm for Twitter Sentiment Analysis and Its Implementation in Mapreduce," 2014. [Online]. Available: https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/45675/research.pdf?sequence=1.

[5] R. Feldman, "Techniques and Applications for Sentiment Analysis," vol. 56 No.4, pp. 82-89, April 2013.

[6] P. Turney, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised classification of reviews," in *Association for Computatoinal Linguistics (ACL),* Stroudsburg, 2002.

[7] B. Pang and L. Lee, "A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts," in *Association for Computatoinal Linguistics (ACL),* Stroudsburg, 2004.

[8] K. Nygard, "Data Mining and Machine Learning with the Naive Bayes Method," Fargo, 2016.

[9] S. Sayad, "An Introduction to Data Mining," 2010. [Online]. Available: http://www.saedsayad.com/naive_bayesian.htm. [Accessed 02 April 2017].

[10] Q. Support, "Naive Bayes Classifier," 2015. [Online]. Available: https://documents.software.dell.com/statistics/textbook/naive-bayes-classifier. [Accessed 28 March 2017].

[11] M. Walaa, A. H. Yousef and K. M. Hoda, "Sentiment Analysis Algorithms and Applications: A Survey," *Ain Shams Engineering Journal,* vol. 5, no. 4, p. 1093–1113, 2014.

[12] T. Mullen and C. Nigel, Sentiment Analysis using Support Vector Machines with Diverse Information Sources, vol. 4, EMNLP, 2004.

[13] R. Berwick, "An Idiot's guide to Support vector," 2003. [Online]. Available: http://www.svms.org/tutorials/Berwick2003.pdf. [Accessed 30 March 2017].

[14] B. Yuan, "Sentiment Analysis Of Twitter Data," Rensselaer Polytechnic Institute, New York, 2016.

[15] C. D. Manning and H. Schuetze, "Foundations of Statistical Natural Language Processing," MIT Press, Cambridge, 1999.

[16] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," in *EMNLP '02 Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, Philadelphia, 2002.

[17] A. Agarwal, B. Xie, I. Vovsha, O. Rambow and R. Passonneau, "Sentiment Analysis of Twitter Data," *LSM '11 Proceedings of the Workshop on Languages in Social Media*, pp. 30-38, 23 June 2011.

[18] A. Agarwal and J. Sabharwal, "End-to-end sentiment analysis of Twitter data," in *Information Extraction and Entity Analytics on Social Media Data*, 2012.

[19] L. Barbosa and J. Feng, "Robust sentiment detection on twitter from biased and noisy data," in *23rd International Conference on Computational Linguistics*, 2010.

[20] S. K. Ravindran and V. Garg, Mastering Social Media Mining with R, Birmingham: Packt Publishing Ltd., 2015.

[21] "Cornell CIS Computer Science," [Online]. Available: https://www.cs.cornell.edu/courses/cs578/2003fa/performance_measures.pdf. [Accessed 24 March 2017].

[22] "KDnuggets," [Online]. Available: http://www.kdnuggets.com/faq/precision-recall.html. [Accessed 25 March 2017].

[23] "Measure," OpenML, [Online]. Available: https://www.openml.org/a/estimation-procedures/1. [Accessed 23 March 2017].

[24] N. M. Sharef, H. M. Zin and S. Nadali, "Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data," *Journal of Computer Sciences*, 2016.

[25] C. D. Manning, P. Raghavan and H. Schütze, Introduction to information, vol. 1, Cambridge: Cambridge University Press, 2008.

[26] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques".

[27] "Multi-Perspective Question Answering," [Online]. Available: http://mpqa.cs.pitt.edu/lexicons/subj_sense_annotations/.

[28] K. Alistair and D. Inkpen, Sentiment Classification of Movie Reviews Using Contextual Valence Shifters, vol. 22, Ottawa, ON, 2006, pp. 110-125.

[29] Z. Li, "Naive Bayes Algorithm For Twitter Sentiment Analysis And Its Implementation In Mapreduce," 2014.