Effective Text Data Preprocessing Technique for Sentiment Analysis in Social Media Data

Saurav Pradha
School of Computing and Mathematics
Charles Sturt University
Melbourne, Victoria, Australia
saurav.pradha54@gmail.com

Malka N. Halgamuge, Senior Member, IEEE
Dep. of Electrical and Electronic Engineering
The University of Melbourne
Victoria 3010, Australia
malka.nisha@unimelb.edu.au

Nguyen Tran Quoc Vinh
Faculty of Information Technology
The University of Da Nang - University of Science and Education, Vietnam
ntquocvinh@ued.udn.vn

***Abstract* ---- In the big data era, data is made in real-time or closer to real-time. Thus, businesses can utilize this ever-growing volume of data for the data-driven or information-driven decision-making process to improve their businesses. Social media, like Twitter, generates an enormous amount of such data. However, social media data are often unstructured and difficult to manage. Hence, this study proposes an effective text data preprocessing technique and develop an algorithm to train the Support Vector Machine (SVM), Deep Learning (DL) and Naïve Bayes (NB) classifiers to process Twitter data. We develop an algorithm that weights the sentiment score in terms of weight of hashtag and cleaned text. In this study, we (i) compare different preprocessing techniques on the data collected from Twitter using various techniques such as (stemming, lemmatization and spelling correction) to obtain the efficient method (ii) develop an algorithm to weight the scores of the hashtag and cleaned text to obtain the sentiment. We retrieved *N=1,314,000* Twitter data, and we compared the popularity of two products, *Google Now* and *Amazon Alexa*. Using our data preprocessing algorithm and sentiment weight score algorithm, we train SVM, DL, NB models. The results show that stemming technique performed best in terms of computational speed. Additionally, the accuracy of the algorithm was tested against manually sorted sentiments and sentiments produced before text data preprocessing. The result demonstrated that the impact produced by the algorithm was close to the manually annotated sentiments. In terms of model performance, the SVM performed better with the accuracy of 90.3%, perhaps, due to the unstructured nature of Twitter data. Previous studies used conventional techniques; hence, no precise methods were utilized on cleaning the text. Therefore, our approach confirms that proper text data preprocessing technique plays a significant role in the prediction accuracy and computational time of the classifier when using the unstructured Twitter data.**

Keywords: *Social media data, Twitter, big data, text data preprocessing, sentiment analysis, Deep Learning, Support Vector Machine, Naïve Bayes, Google Now, Amazon Alexa.*

I. INTRODUCTION

Background

One of the most significant current discussions in the world is big data. In recent years, there has been a considerable rise in social media giants such as Twitter thus proving them to be a massive amount of big data [1, 2, 3]. Those data can then be collected in large volume and can then be utilized to train the machine learning and Deep Learning, which will aid in decision making.

Sentiment analysis is one of the methods of extraction of text from various sources for personal or commercial use. Due to the popularity of social media, everyone posts a massive amount of data online, which can then be used to generate sentiments. This can be utilized to give companies an insight into their product. Information such as performance of the products throughout the year, analysis of different competing products and can be extracted and utilized to the company's advantage.

The detection of the real-time abuse of the drug using the tweets has been analyzed by Phan et al. [4]. Authors use legal and illegal drugs dataset, original text with the collection of 31,478 tweets. It does not use any preprocessing and uses the J48, Random Forest, Naïve Bayes, and SVM (Support Vector Machine) classifier for training purposes. The developed classifier developed has been tested on the real-world tweet dataset with the precision of 74.8% with the J48 algorithm. The suggested work includes Term Frequency-Inverse Document Frequency (TFIDF) used to reflect the relevance of the term in the given document and to improve the accuracy and to use Mechanical Turk for the collection of vast amounts of data.

In another study, Bhat et al. [5] used sentiments for the development of the system that observes the opinion by people on some product or people. It uses the Twitter API to extract 1000 latest tweets and performs text processing like stemming and stop word removal. No machine learning algorithm was used for the classifying purpose. This gave us a model that calculates the sentiment by multiplies of adverbs value instead of summing up the whole sentiment of tweets. Future work such as the

978-1-7281-3003-3/19/$31.00 ©2019 IEEE

development of the algorithm for identifying the offensive statements and, improving the efficiency of mapped words are suggested.

It is quite challenging to process unstructured data; hence, social media data is challenging to manage and requires proper preprocessing before obtaining the right sentiment. Past investigations utilized standard techniques for cleaning the text; thus, no appropriate strategies were utilized.

Motivation

Preprocessing method is used to produce the correct sentiment for effective decision making. It is implemented to remove the unstructured nature of data obtained from social media. However, to apply this method to massive datasets will require much time. In this situation, the method that requires computationally less time and more accuracy should be chosen. For the useful generation of the sentiment, the sentiment of the cleaned text and hashtag should be used in combination because hashtag also gives context on the importance of the topic. Furthermore, as the datasets are large, effective classifier with more accuracy and less computational time should also be chosen to obtain a faster result.

Paper Contributions

The main contributions of this paper include the following:

- Develop a new algorithm to provide proportional weight between the hashtag and cleaned text combined to obtain sentiment output.
- Conduct an extensive comparison of the popularity of two products: *Google Now* and *Amazon Alexa* using 1,314,000 unstructured tweets.
- Compare three different types of preprocessing technique (Stemming, Lemmatization and Spelling Correction) and its effect on sentiment produced.
- Compare sentiment provided by the user, algorithm, uncleaned sentiment and the sentiment provided by the cleaned text.
- Compute the computational speed and accuracy of Support Vector Machine (SVM), Deep Learning (DL) and Naïve Bayes (NB) classifiers to assess the better performer.

The paper is structured as follows: Section 2 begins by elaborating different cleaning process and data analysis part, which consists of complete details on the algorithm and classifiers used. Section 3 describes the result of using the method and methodology discussed in Section 2. The details on the improvement of the paper are discussed in Section 4 followed by the conclusion is presented in Section 5.

I. MATERIALS AND METHODS

A. Data Collection

The data was collected using the *GetOldTweets3* Python 3 library. Tweets were collected to observe an extensive comparison of the popularity of two products: *Google Now* and *Amazon Alexa.* The total data that was collected together was *N= 341K* tweets for *Google Now* and *N= 1* million datasets from *Amazon Alexa*. The data contains information about the tweets that the customer tweeted about *Google Now* and *Amazon Alexa*.

The data attributes that are targeted in the collection of the tweets are as follows: original tweets, clean tweets, polarity, sentiments, hashtags, user mention, retweets, favourites, permalink, tweet length, twitter ID.

B. Selection Process of Efficient Data Preprocessing Techniques

Previous studies have used different preprocessing techniques as given in Table 1. Various experiments were carried out with different techniques or combinations of techniques to observe which method would most likely give us a better outcome. Preprocessing was performed using the techniques mentioned in Table 1.

Table 1: Description of the Preprocessing techniques

No	Technique Name	Description
1	**Lower text**	The primary purpose of lowering the text to all lowercase is so that the word such as "Hello" and "hello" would not be treated as a different word since they are the same word. It helps in reducing the number of words that the dictionary needs to hold at a time [6].
2	**Removal of @mention**	It facilitates in the removal of the user mentions as they do not provide any relevant information about the text sentiment.
3	**Removal of URL**	It is done to remove any URL from the tweet. It includes the removal of URLs starting HTTP, https and also pic:\\ (it is the URL for the picture in the tweet)
4	**Removal of punctuation**	Removing the punctuation or any non-alphanumeric words from the original text.
5	**Removal of the hashtag**	The hashtag is removed from the text. For this process, the hashtag is removed from the text and stored in a separate column. Then we can use the weight of the hashtag in a separate process.
6	**Removing whitespace**	Whitespace does not provide any meaning to the text, so it is removed for computational purposes.
7	**Tokenization**	It is the splitting of each sentence to text.
8	**Removal of the encoded text formats**	For this study, the encoded texts are removed, and only the ones that give specific meaning are kept. Example of it includes removal of words such as xbf, x9a etc.
9	**Stop word removal**	Words such as "a", "an", "the" do not provide any meaning to the text. Hence, those words have been removed. Now the relevant text is used for the sentimental purpose. It builds the exactness of the text [7].
10	**Stemming**	Stemming is the conversion of the words to their root meaning. It makes lessens the total words and facilitates in the computational speed.

B.1 Techniques Comparison:

A variety of methods are used to assess the preprocessing of the text. Their descriptions are given in Table 2:

Table 2: Data Preprocessing Techniques used in this study

Techniques	Description
Technique 1 (Stemming)	It is the stemming process. It is where the words are converted to their root words, which makes the length of the words smaller. It thus facilitates the computation process.
Technique 2 (Lemmatization)	It is the lemmatization. It is like the stemming process and returns the base or dictionary form of the word
Technique 3 (Spelling Correction)	It is the usage of the lemmatization and addition spelling correction. The words are first passed for the spelling correction, where the spelling is checked and corrected if necessary. Then it is passed on to lemmatization.

To use three different techniques, first, the data was passed to the common process, which consists of techniques mentioned in Table 1.

The sentiment obtained was then stored in the list. In the same time, the sentiments were converted to the numeric value by finding out which type of sentiment:

$$f(x) = \begin{cases} 0 & if\, x = \text{negative} \\ 1 & \text{if } x = \text{neutral} \\ 2 & \text{if } x = \text{positive} \end{cases}$$

where x is the sentiment (positive, negative or neutral).

In order to identify the performance of the technique, the steps presented in Figure 1 was used.

C. Data Analysis

Data analysis was run on *Lenovo G50* with the i7 processor running latest *Windows 10* computer system. Appropriate libraries were installed to run the proposed programs. The data have been gathered by using a python library named *GetOldTweets3*. That data was preprocessed and then passed to the algorithm developed. The algorithm uses the weight of the hashtag and the weight of the cleaned tweet. The purpose of using the hashtag weight is because it is useful in a recommendation system, classification, categorization, search. Hashtags (i) facilitate the search of topics based on social content with themes, (ii) provide support to the user to identify relevant topics, (iii) the recommendation system built using hashtag has also received much attention. All of which highlights the importance of using hashtags in our algorithm.

D. Proposed Algorithm

We develop a new algorithm to provide proportional weight between the hashtag and cleaned text combined to obtain sentiment output.

The weight is calculated using

$$Tw = \alpha H + \beta T \qquad (1)$$

where $\boldsymbol{\alpha}$ is the weight of the hashtag and $\boldsymbol{\beta}$ is the weight of the tweets, $\boldsymbol{H}$ is the hashtag score, and $\boldsymbol{\beta}$ is the clean tweet score.

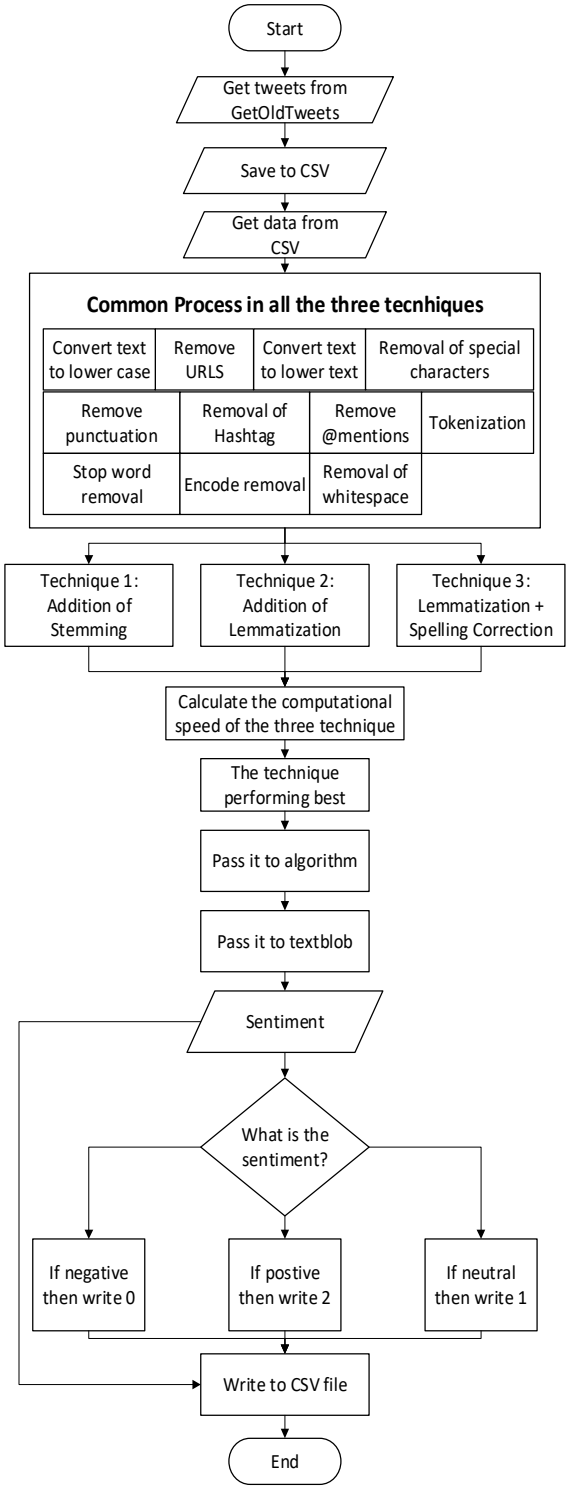

Figure 1: Overall selection process of the best performing preprocessing techniques

Algorithm for weighting the text:

```
While (tweet != 0):
        Calculate the tweet sentiment polarity
    If (hashtag != 0 ):
        Calculate the hashtag sentiment polarity
    End if
    If (hashtag == 0):
        Set the weight of the tweets to 100%.
    End if
    Repeat:
        Compute Tw=αH + βT
        Assign the final polarity
        Based on polarity determine the sentiments
Until there are no tweets left
```

The detailed flow of the algorithm is shown in Figure 2. After the successful selection of the preprocessing technique, it is passed to the algorithm to generate sentiment. The algorithm works on the weight of text and the weight of the hashtag. Hashtags indicate the main subject of the text, has been shown to provide valuable information and has been used in sentimental analysis. For example, the proportional weight given to the hashtag in the algorithm is 40%, and the weight given to the cleaned text is 60%. If there is no hashtag in the tweet, the full weight (100%) is given to the cleaned text, and the sentiment is produced.

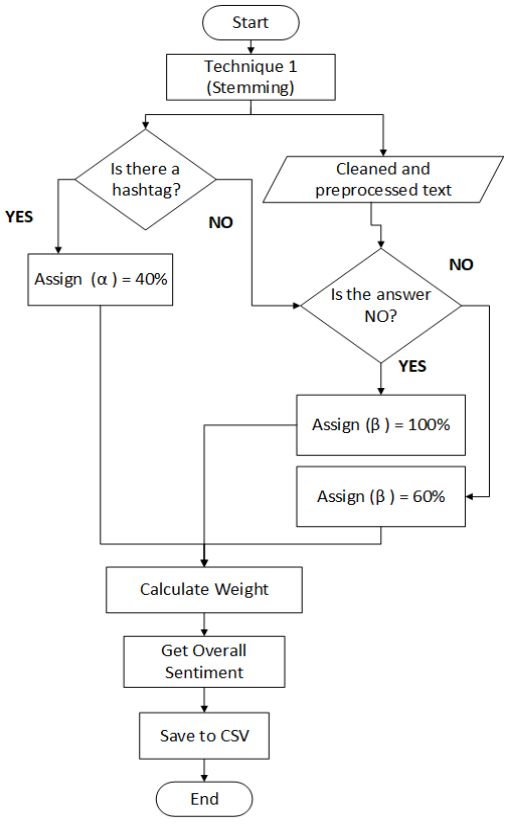

Figure 2: Proposed algorithm after the selection of the preprocessing technique

II. Results

Techniques Comparison:

Figure 3 shows the result of sentiments given by three techniques. The result of those techniques is then compared against the clean sentiment tweets.

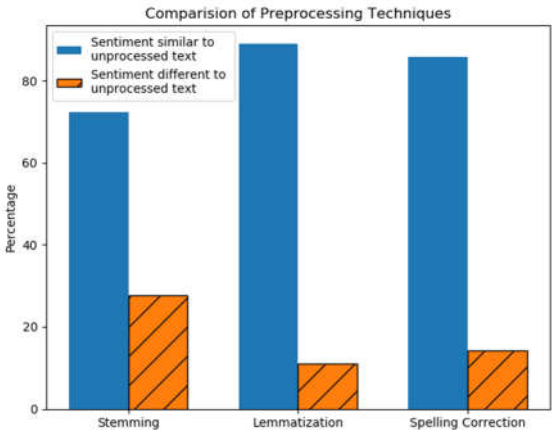

Figure 3: Performance of the three cleaning techniques with the unprocessed sentiment text

The result of comparing the three techniques with the sentiment of the uncleaned or unprocessed text is shown in Figure 4**.** Each sentiment given by three different techniques is compared with the unprocessed text. As indicated in Figure 4, the rate of dissimilarity was 28% compared to the other two (Technique 2 and Technique 3), which was 11 and 14 %. It shows that the result produced by Technique 1 is more different compared to the unprocessed sentiments.

Techniques Comparison:

Table 3: Technique Speed Comparison

Technique	Speed as Percentage
Stemming	0.25
Lemmatization	0.53
Spelling Correction	100.00

As Table 3 illustrates, there is a significant difference between Technique 3 and the other two technique. Technique 3 takes nearly 400 times more time than Technique 1, as Technique 3 (spelling correction) goes through each word in a text and matches whether it corrects spelling or not. If the spelling is not correct, it fixes the spelling. This results in requiring a considerable amount of time for massive sets of data.

Figure 4 shows the visualized form of Table 3.

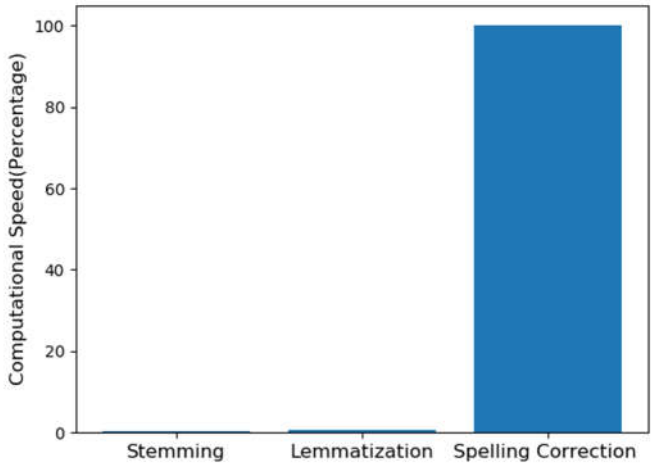

Figure 4: Computational Speed in the second visualization of three techniques

It is evident that Technique 1 requires less computational speed as illustrated by Figure 4 and gives different sentiment than the unprocessed text as evident from Figure 3.

Comparison of Different Sentiments:

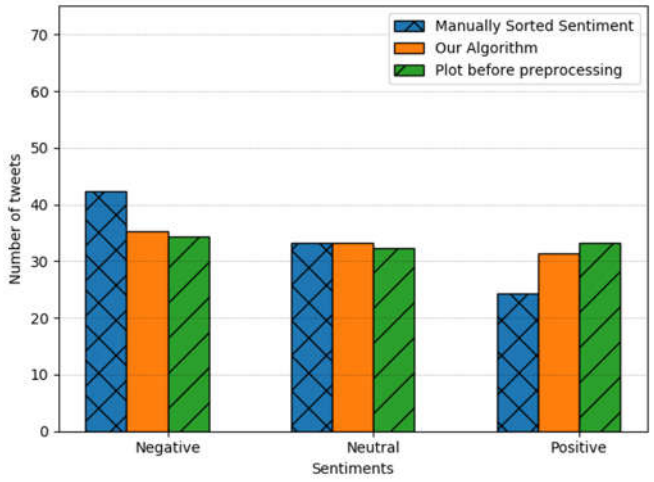

Figure 5: Comparison of Negative, Neutral and Positive Sentiment produced by three different methods

Figure 5 illustrates the different types of sentiment produced by the user, such as a sentiment produced by an algorithm, sentiment generated without cleaning the data and sentiment produced after cleaning data. This demonstrates our algorithm significantly accurate than all other cleaning processes as compared to the manually sorted sentiment.

Classifiers Performance:

The sentiment produced by the algorithm is used to train the classifiers. Computation speed has been calculated based on both the training phase and the decision phase. The computational speed and accuracy from those classifiers are presented in Table 4.

Table 4: Accuracy of Three Classifiers on a random set of N = 10,000 datasets from 1 Million data for classifier train

Classifiers	**Accuracy (%)**	**Computational Speed (sec)**
Deep Learning	70.96 (epoch = 10)	224
Naïve Bayes	65.09	752.77
Support Vector Machine	90.3	142.64

Note that the speed calculation is based on the time required to train the classifier. Support vector machine outperformed both the Naïve Bayes and Deep Learning with the accuracy of 90.3% when used the random tweets (*N =10,000*) of 1 million retrieved tweets. The time required to train the classifier was around 142.54 seconds. The reason for taking random sets of tweets 10,000 is due to the difficulty of obtaining a large quantity of manually annotated tweets.

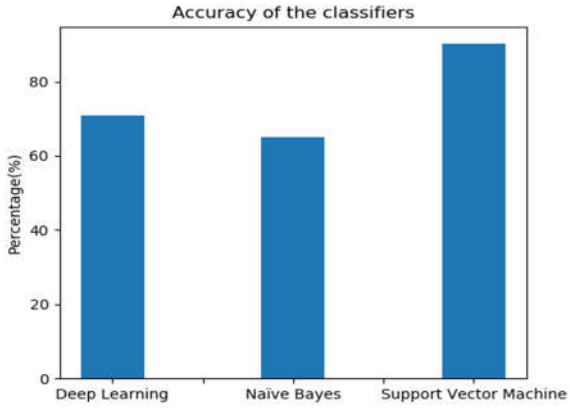

Figure 6: The accuracy of the three classifiers against the amount of computational time in seconds.

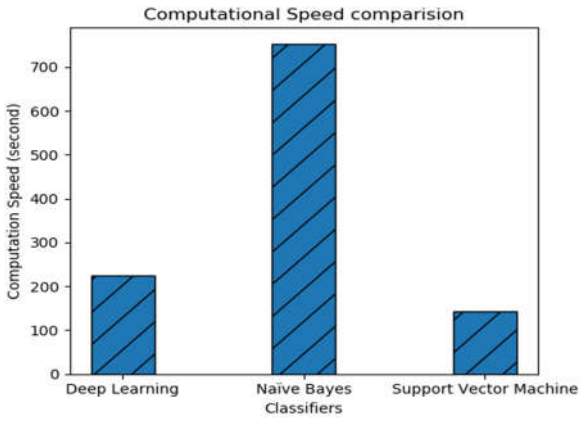

Figure 7: The computational speed of the three classifiers against the amount of computational time in seconds.

Figure 5 illustrates the computational speed of the classifiers with accuracy. The accuracy as a percentage while the computational speed is measured in terms of speed in second. Here, the least time is taken by Support Vector Machine with more accuracy than any other classifier.

Overall Sentiments throughout the years:

(a) Google Now:

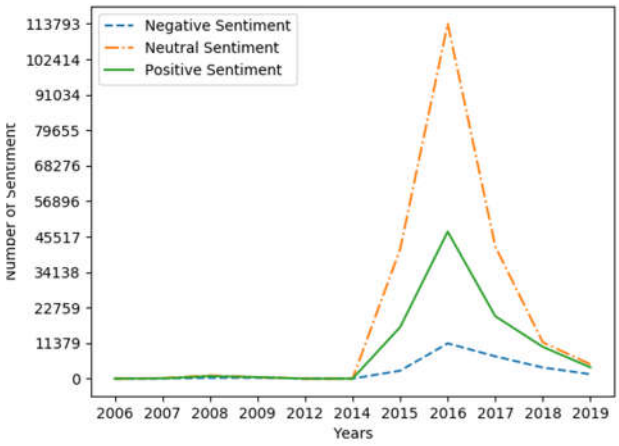

Figure 8: Overall Sentiments curve of Google Now from the start of 2006 to 2019 on N = 341K sets of tweets.

Please note that this sentiment is based on the number of tweets that are collected from a specific time. The total tweet collected were *N = 341K* tweets from 2006 to 2019. The data that was collected from 2019 was from January to the 1st of May 2019.

As shown in Figure 8, the peak time for *Google Now* was in 2016 with a slow decline after that year. The reason for the peak in 2016 is because Google released a daydream VR (virtual reality) platform, which utilized *Google Now*. In addition, there was a major changed to Google as they launched Google Assistant – a next big evolution to *Google Now*.

(b) Amazon Alexa:

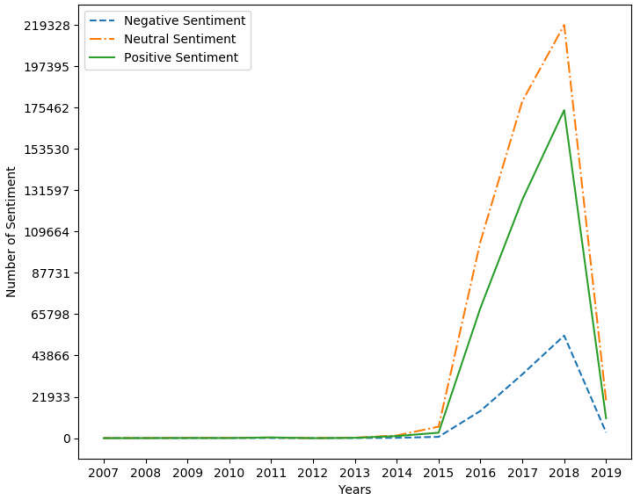

Figure 9: Overall Sentiment of Amazon Alexa on the year-wise basis

The release date of *Amazon Alexa* was in 2014. Since then, Amazon has launched Amazon Echo in 2017, which is using the *Amazon Alexa* voice assistants. Because of this, the popularity of *Amazon Alexa* and its hashtag #alexa has increased a lot. As illustrated from Figure 9, 2018 was the most popular year for the *Amazon Alexa*. It elaborates that 2018 was the year with the most tweet and most popularity.

Comparison of *Amazon Alexa* and *Google Now* for the Year 2016:

(a) Google Now:

As seen from Figure 8, the peak time for *Google Now* was in 2016. Monthly sentiment values were observed to find out the popularity of *Google Now*.

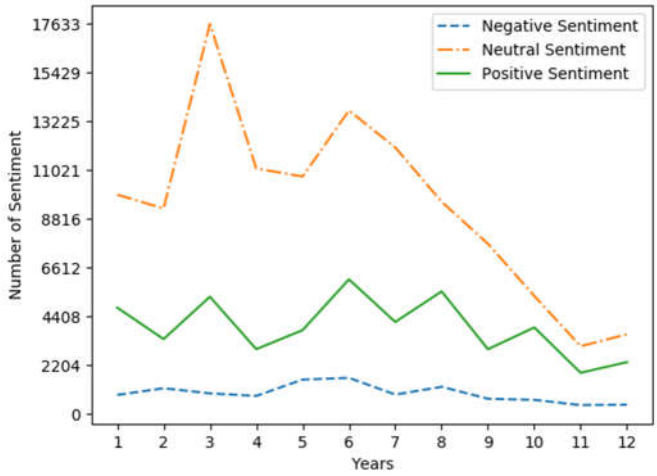

Figure 10: Overall Sentiment curve of Google Now monthly for the Year 2016.

Figure 10, demonstrates the peak time for *Google Now* in 2016 was in March with the highest scaling.

(b) Amazon Alexa:

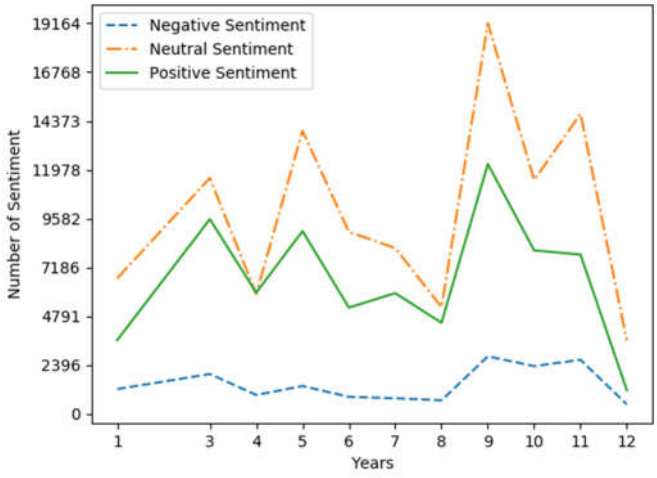

Figure 11: Overall Sentiment curve of Amazon Alexa monthly for the Year 2016.

Similarly, in Figure 11, it is also shown that September was the most popular month for the *Amazon Alexa*.

Comparison of *Amazon Alexa* and *Google Now* for the Year 2018:

(a) Google Now:

We observed this relationship from the recent data, 2018 to compare the popularity of *Google Now* and *Amazon Alexa*. Figure 12 illustrates the sentiment values for *Google Now* monthly. In contrast to 2016 data, it is evident that the best month for *Google Now* in 2018 is on November.

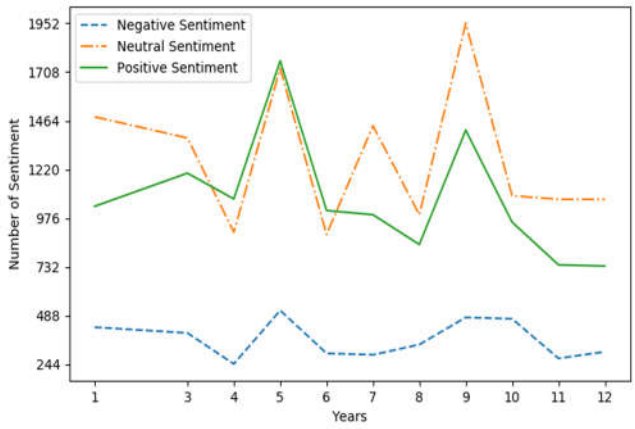

Figure 12: Overall Sentiment of Google Now in month wise basis for the year 2018

(b) Amazon Alexa:

Similarly, in contrast to 2016 data, it is evident that the best month for *Amazon Alexa* in 2018 is on November.

The most popular month for *Amazon Alexa* was January 2018 with the overall tweets of 66,081. Figure 13 presents the sentiment values *for Amazon Alexa* monthly. It shows that the most popular month for *Amazon Alexa* was on January using 66081 total tweets.

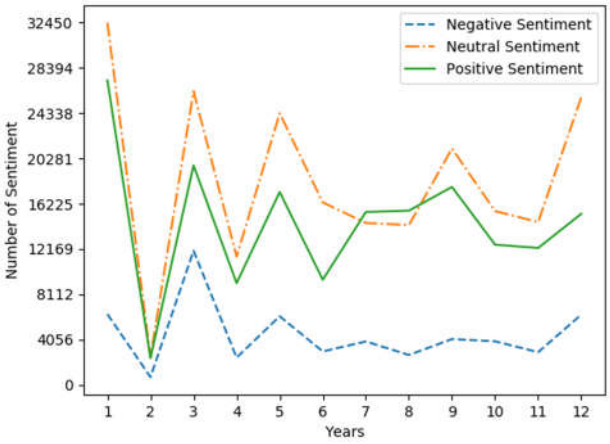

Figure 13: Overall Sentiment of Amazon Alexa in month wise basis for the year 2018

Overall Popularity:

Figure 8 and 9 demonstrate that the most successful year for *Google Now* was 2016 with *N = 172,240* tweets and the most successful year for *Amazon Alexa* was 2018 with *N = 447,911* tweets. For *Google Now*, the most popular month for 2016 was March with a total tweet of *N = 23,882* tweets, as shown in Figure 10. For *Amazon Alexa*, the most popular month in 2018 was January with total tweets of N = 66,081, as shown in Figure 13.

4. DISCUSSION

Sentiment analysis using freely available Twitter data is quite essential to gain an insight into a product or to observe the popularity of a product or organization. This information helps the business to make a better-informed decision and help companies to make a considerable profit from it. Effective text preprocessing is required, hence, it can generate accurate sentiments. Also, the nature of tweets being the short length, people use the shortcut language, and short abbreviations in their tweets which makes it crucially important to preprocess the text effectively otherwise the sentiment generated from short tweets will also be wrong.

Some studies have used only simple programming for preprocessing, while others have used standard techniques like tokenization, stemming from obtaining a better result. In our study, the combination of the preprocessing techniques such as stemming, tokenization, removal of special character, punctuation, and usage of simple programming combined performed better in the cleaning process. Hence, our results show that the data works well with a classifier such as Naïve Bayes, Deep Learning and with the SVM classifier giving the higher result with less computational speed to obtain the result. In contrast [8], the Naive Bayes and Random Forest classifiers are more susceptible than Logistic Regression and support vector machine classifiers when different pre-processing techniques were utilized. Choosing a learning algorithm relies upon the elements of the application [9]. Expanding the number of features embraced by classifier furthermore builds the element space dimension causing "curse of dimensionality". This makes learning confused with less accuracy and more substantial computation time.

There are obvious limitations to our study. In general, sarcastic tweets include positive words or even increased positive words to pass on a negative opinion or the other way around. This decreases the accuracy of the classifier as it will classify those text to the wrong sentiments. Moreover, the scenario where the tweet only contains a picture, links or mentions is also excluded. In our study, there were multiple instances of that tweet which had to be discarded and will be included in the future improvements.

The result obtained from this study confirms that proper preprocessing technique plays a significant role in cleaning the text and increasing the accuracy of the classifier. In terms of choosing classifiers, Naïve Bayes and SVM were the most common classifiers used and with them performing better among all the classifier. Even among those two, SVM performed better in most of the cases.

III. Conclusion

In recent years, there has been a considerable rise in social media data such as Twitter which proves that they are a vast amount of big data that could be utilized for the decision-making process. Nonetheless, those data are unstructured. Text data preprocessing is one of the effective methods in terms of cleaning and making those unstructured data, structured and meaningful. We have compared three different types of text data preprocessing technique (Stemming, Lemmatization and Spelling Correction) and its effect on sentiment produced. Our algorithm can be utilized to provide proportional weight between the hashtag and cleaned text combined to obtain sentiment output. First, three different preprocessing methods were compared to determine the best performing method, and then the result was passed to an algorithm developed. So proper sentiments can be made. These sentiments were then used for training the classifiers (SVM, NB, and DL). Our analysis shows the SVM performed better with the SVM algorithm, perhaps due to the unstructured nature in Twitter data. Moreover, the evidence from this study suggests the implications of choosing the correct text data preprocessing on sentiment produced will facilitate quick and accurate decision making. Additionally, correct sentiments can be used to map the overall sentiments produced throughout the year to perceive the popularity of products and obtain insight into its overall performance. This information can then be utilized by businesses to make a profitable and better-informed decision in the future.

References

[1] M. Khader, A. Awajan and G. Al-Naymat, "The Effects of Natural Language Processing on Big Data Analysis: Sentiment Analysis Case Study", 2018 International Arab Conference on Information Technology (ACIT), 2018.

[2] A. Singh, M. N. Halgamuge, and B. Mouess, "An Analysis of Demographic and Behaviour Trends using Social Media: Facebook, Twitter and Instagram", Social Network Analytics: Computational Research Methods and Techniques, Elsevier, Chapter 5, ISBN: 9780128154588, January 2019.

[3] S. Kalid , A. Syed, A. Mohammad, and M. N. Halgamuge, "Big-Data NoSQL Databases: Comparison and Analysis of "Big-Table", "DynamoDB", and "Cassandra", IEEE 2nd International Conference on Big Data Analysis (ICBDA'17), Beijing, China, pp 89-93, 10-12 March 2017.

[4] N. Phan, S. Chun, M. Bhole and J. Geller, "Enabling Real-Time Drug Abuse Detection in Tweets", IEEE 33rd International Conference on Data Engineering (ICDE), 2017.

[5] S. Bhat, S. Garg and G. Poornalatha, "Assigning Sentiment Score for Twitter Tweets", 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2018.

[6] L. Batista and L. Alexandre, "Text Pre-processing for Lossless Compression", Data Compression Conference, Dhaka, Bangladesh, 2008.

[7] B. Savaliya and C. Philip, "Email fraud detection by identifying email sender", 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017.

[8] J. Zhao, and G Xiaolin, "Comparison research on text pre-processing methods on twitter sentiment analysis." IEEE Access, Vol 5, pp 2870-2879, 2017.

[9] A. Singh, M. N. Halgamuge, R. Lakshmiganthan, "Impact of Different Data Types on Classifier Performance of Random Forest, Naïve Bayes, and k-Nearest Neighbors Algorithms", International Journal of Advanced Computer Science and Applications (IJACSA), Vol 8, No 12, pp 1-10, December 2017.