I.J. Intelligent Systems and Applications, **2021, 4, 1-13**
Published Online August 2021 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2021.04.01

Sentiment Analysis on Twitter Data: Comparative Study on Different Approaches

[1]Abdur Rahman, [2]Mobashir Sadat, [3]Saeed Siddik
[1]Centre for Advanced Research in Sciences, [2,3]Institute of Information Technology
University of Dhaka, Dhaka, Bangladesh
E-mail: {[1]mukul.arahman, [2]sadat.mobashir}@gmail.com, [3]saeed.siddik@iit.du.ac.bd

Received: 08 January 2021; Revised: 15 March 2021; Accepted: 29 April 2021; Published: 08 August 2021

Abstract: Social media has become incredibly popular these days for communicating with friends and for sharing opinions. According to current statistics, almost 2.22 billion people use social media in 2016, which is roughly one third of the world population and three times of the entire population in Europe. In social media people share their likes, dislikes, opinions, interests, etc. so it is possible to know about a person's thoughts about a specific topic from the shared data in social media. Since, twitter is one of the most popular social media in the world; it is a very good source for opinion mining and sentiment analysis about different topics. In this research, SVM with different kernel functions and Adaboost are experimented using CPD and Chi-square feature extraction techniques to explore the best sentiment classification model. The reported average accuracy of Adaboost for Chi-square and CPD are 70.2% and 66.9%. The SVM radial basis kernel and polynomial kernel with Chi-square n-grams reported average accuracy of 73.73% and 68.67% respectively. Among the performed experimentation, SVM sigmoid kernel with Chi-square n-grams provided the maximum accuracy that is 74.4%.

Index Terms: Sentiment Analysis, Machine Learning, Twitter Data Comparative Analysis.

1. Introduction

Sentiment Analysis is a natural language processing scheme used to determine whether a data is positive or negative [1,2]. This allows determining people's attitude and opinion in relation to different topics, products, events, etc. The role of sentiment analysis has been growing significantly with the rapid spread of social networks. Twitter is one of the most popular social network applications. People from all over the world use twitter. Through the twitter platform, users share information, opinion about politicians, products, companies, events, etc. For this reason, twitter has been attracting the attention of different communities interested in analyzing its content. Researchers have used twitter data in making different predictions such as box office revenue of movies, election results etc. However, the most important step in making those predictions is sentiment analysis with shared tweets regarding the topic about which prediction was being made.

Sentiment analysis is a time consuming task because of the huge volume of tweets and the volume is increasing continuously. To get a general idea of public sentiment towards a topic, it is not possible to analyze the sentiment of every tweet manually about that topic. To this end, an automated system is required which can predict a tweet's sentiment. Researchers used different approaches in analyzing sentiment analysis of tweets.

Pak et al. built a sentiment classifier for tweets using the naïve bayes model [3]. Wang et al. presented a real-time public sentiment analysis model toward presidential candidates in the 2012 US election [4]. Kouloumpis et al. experimented twitter sentiment analysis using Adaboost algorithm and linguistic feature extraction techniques [5]. Silva et al. illustrated sentiment analysis using ensemble learning method named as adaboost with different feature extraction techniques for tweets [6]. Barbosa et al. used Support Vector Machine (SVM) to classify tweets from twitter according to their sentiment [7]. Elzayady et al. demonstrated both machine and deep learning techniques for sentiment analysis on Arabic reviews [8].

In the literature, different approaches were taken in selecting features and then different algorithms were used for training models to analyze sentiment analysis of tweets. However, Categorical Proportional Difference (CPD) and Chi-square based feature extraction for sentiment analysis is rarely found, which are good candidates for supervised text classification [11, 12]. The main objective of this study is finding the best SVM kernel function with optimal feature selection strategy. We have experimented with radial basis, sigmoid and polynomial SVM kernels and Adaboost to find the best twitter sentiment classifier. Furthermore, performance of classifiers depends on feature selection accuracy, where we have considered CPD and Chi-square strategies.

A pre-annotated dataset containing 1,578,627 classified tweets was collected to be applied on this experimentation.

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

The dataset is pre-processed to eliminate unnecessary text and improve accuracy of the models. Next, textual features are applied on constructed tweeter classification models. This is a very important step where four feature extraction techniques named as n-grams, lexicon, emoticon and intensifiers are considered to train the model. The n-gram features were selected with categorical proportional difference and Chi-square strategy. The value of both the techniques is calculated for each n-gram in the tweets.

After getting features, Adaboost and SVM were used as models to classify the tweets as positive or negative. SVM classifiers were built using (i) radial basis (ii) sigmoid and (iii) polynomial kernels for both Chi-square and CPD feature selection techniques.

According to the reported results, precision and accuracy of Chi-square is always higher than the CPD for all considered classification algorithms. The average accuracy of Adaboost for Chi-square and CPD were 70.2% and 66.9% respectively.

On the other hand, the average accuracy of the SVM sigmoid kernel with Chi-square n-grams is 74.4%. Where, the average accuracy of SVM radial basis kernel and polynomial kernel with Chi-square n-grams were 73.73% and 68.67% respectively. So, this experiment shows that the sigmoid kernel of SVM fits the dataset and best followed by radial basis kernel and polynomial kernel. Therefore, it is concluded that n-grams selection with Chi-square is a better feature selection technique than n-grams selection with CPD value. And the SVM sigmoid kernel for n-grams selected with Chi-square performed the best.

The main contributions of this research is to (i) empirically investigate the tweet classification performance of Adaboost and SVM kernels for CPD and Chi-square feature selection techniques, and (ii) identifying the best combination of feature selection and classification method.

The rest of this paper is organized as follows: section II discussed the related work, section III is about the proposed methodology, and section IV illustrates the result of the analysis. Finally, section V concludes with a future research scope.

2. Related Work

With the population of blogs and social networks, sentiment analysis became a field of interest for many researchers. This is a growing research area of Natural Language Processing (NLP). In this section, twitter sentiment analysis and classification researches have been discussed.

Shaheen et al. analyzed unlocked mobile reviews to find out interesting facts, trends, figures and the relationship among different attributes of reviews such as length, number of review ratings and price [1]. In addition, sentiment analysis on the dataset also performed to formally compare the performance of implemented classification algorithms. Seven different algorithms named as gradient boosting, stochastic gradient descent, multinomial naïve bayes, long short-term memory, random forest, NB-support vector machine, and convolutional neural networks were applied for review classification. The amazon.com mobile reviews dataset available on kaggle.com was used for experimentation. The research work showed different correlations among reviews attributes.

The analysis showed acceptable positive correlation between longer reviews and helpfulness. However, there is weak correlation between review length and product price. The research considered that correlation exists among reviews, product price and product rating. The most prevalent positive and negative words over the dataset are great, good, camera, price, excellent and return, back, problem, charge respectively. The sentiment score for eight emotions named as anger, anticipation, disgust, fear, joy, negative, positive, sadness, surprise and trust were reported. According to the results, random forest outperformed remained classifiers and shows 85% accuracy for the given reviews dataset.

Pak et al. built a sentiment classifier for tweets using the naïve bayes classifier [3]. The authors showed how to collect a corpus for sentiment analysis automatically and built a sentiment classifier that is able to determine positive, negative and neutral sentiments for a tweet. 300000 tweets were collected using Twitter API where emoticons acted as filter parameters. The happy emoticons like ":-)", ":)", "=)", ":D" etc. and sad emoticons like ":-(", ":(", "=(", ";(" etc. were used to crawl positive and negative sentiment tweets respectively. Where, the tweets without any emotion were considered as neutral. The corpus of objective tweets i.e. neutral tweets was retrieved from Twitter users of popular newspapers and magazines such as "New York Times", "Washington Posts" etc. The work used n-gram and part of speech (POS) as feature extraction technique and naive bayes as classifier. The classifier was trained with unigram, bigram and trigram features. It was tested out on a hand annotated dataset of real twitter posts.

The F-measure showed that the performance of the classifier built with bigrams was best. However, the study did not consider the noise in the datasets and also did not remove the stop words in the tweets. The work removed the n-grams, which did not contribute to any sentiment by finding their entropy but that seems like an extra step.

Wang et al. presented a real-time public sentiment analysis model toward presidential candidates in the 2012 US election [4]. All relevant tweets in real-time were collected from the entire Twitter traffic via Gnip Power Track, a commercial Twitter data provider. The rules were constructed manually that are simple logical keyword combinations to retrieve relevant tweets about each candidate. The training data consisted of nearly 17000 tweets (16% positive, 56% negative, 18% neutral, 10% unsure) were labelled by 800 Turkers. Unigram features were extracted based on tokenization of the tweets that attempts to preserve punctuation that may signify sentiment. The classifier was built

using naïve bayes model with unigram features.

The classifier performed at 59% accuracy on the four category classification of negative, positive, neutral, or unsure. The implemented methodology was a very useful one, where the architecture and method are generic, also can be easily adopted and extended to other domains.

Kouloumpis et al. experimented twitter sentiment analysis using adaboost machine learning algorithm and linguistic feature extraction techniques [5]. The tweets were classified into positive, negative and neutral classes for three different corpora of twitter messages. Three features were extracted using prior polarity: positive, negative and neutral. Parts of Speech or POS features such as count of verbs, adjective, nouns were extracted for each tweet. Microblogging features were created as binary features which capture positive, negative and neutral emoticons, abbreviations and the presence of intensifiers. The AdaBoost models with 500 rounds of boosting were performed to train the classifier. The hashtagged data set compiled from the edinburgh twitter corpus and emoticon dataset created by Go, Bhayani, and Huang for a project at Stanford University were used to train the classifier. The classifier was trained several times using n-gram features, n-gram+lexicon features, n-gram+lexicon+microblogging features, and all features.

The experimented result showed that POS features are not useful and microblogging features such as positive/negative emotions are useful for sentiment analysis. The best F-measure value was 0.68. However, the approach did not consider noise in the training data. Also, it was ignored that a tweet with a positive emoticon or hashtag does not necessarily have a positive sentiment.

Silva et al. also evaluated sentiment analysis using ensemble learning method named as adaboost with different feature extraction techniques for tweets [6]. Three types of features named as lexicon features, feature hashing and POS tagging were extracted to train the classifier. Sentimental lexicon known as SentiStrength was used, which provides an emotion vocabulary, an emoticons list (with positive, negative, and neutral icons), a negation list, and a booster word list. For each tweet, positive, negative and a neutral score feature was extracted. The score was assigned based on the emotion vocabulary, emoticon list, negation list and the booster word list. The feature hashing offers an approach to reduce the number of features provided as an input to a learning model. The MurmurHash3 function was used to reduce the number of unigram features extracted from the dataset. Multiple words are mapped to a key which represents the features mapped to the same key. The total number of keys is 1024. The extracted unigrams are mapped to one of those keys. On the other hand, POST tagging was performed with the Twitter NLP tool. It encompasses 25 tags including nominal, nominal plus verbal, other open class words like adjectives, adverbs and interjection, Twitter specific tags such as hashtags, mention etc.

The experimentation was performed on WEKA platform where AdaBoost with Multinomial Naive Bayes (MNB) and support vector machine was applied as a classifier. The research tested out their test-set with classifiers trained with AdaBoost, MNB and support vector machine. The best result in terms of accuracy was found when the tweets were classified using adaboost with MNB as the component.

Barbosa et al. classified tweets textual data using support vector machine according to their sentiment [7]. The proposed technique followed a 2-step sentiment analysis method, where first classifies messages as subjective and objective, and further distinguishes the subjective tweets as positive or negative. PoS features, prior subjectivity (weak and strong subjectivity) and prior polarity (positive, negative, neutral) of words were collected using subjectivity lexicon[1] of 8000 words. In addition, common slangs used in twitter were added to the lexicon. SVM algorithm was applied on the WEKA platform to train the classifier, where polarity, subjectivity, PoS, link and upper case were considered as features. The TwitterSA (cleaning), TwitterSA (no-cleaning), TwitterSA(voting) and TwitterSA (maxconf) classifiers were used.

The data has been collected from three commercial sources named as Twendz, Twitter Sentimentand, and TweetFeel. TwitterSA (cleaning) performed the best in subjectivity classification with an error rate of 18.1%. For polarity classification, TwitterSA (maxconf) gave the best result with an error rate of 18.7%.

Elzayady et al. proposed both machine and deep learning methods for sentiment analysis on Arabic reviews data [8]. Three machine learning algorithms named as naïve bayes, k-nearest neighbor and decision trees were considered in this research. Where, Long Short Term Memory (LSTM) and Convolutional Neural Networks (CNN) were used as deep learning classifiers. In addition, a model was proposed combining CNN-LSTM architecture where output of CNN was fed as input of LSTM. Both the deep learning approaches named as CNN and RNN outperformed the applied machine learning approaches. However, the combined CNN-LSTM architecture shows impressive results with average accuracy of 85.83%, 86.88% for Hotel Reviews (HTL) and Book Reviews (LABR) datasets respectively.

Asur et al. forecast box-office revenues for movies using linear regression model for tweets [9]. This paper showed that a simple model built from the rate at which tweets are created can predict the box office revenue of movies. The tweets about a movie can be both positive and negative. Tweets with positive sentiment will encourage more people to go see the movie, on the other hand, tweets with negative sentiment may discourage people from watching it. So, the box office performance of a movie depends not only on the tweet rate about a movie but also on what is being said in those tweets.

So, a sentiment analysis was performed using n-gram features and the tweets were divided into positive, negative and neutral classes. A linear regression model was built to predict the box office revenue of a movie at a given weekend.

[1] The subjectivity lexicon is available at http://www.cs.pitt.edu/mpqa

The dataset was obtained by crawling hourly feed data from Twitter. The prediction of box office revenue was calculated and the predictions had an R squared value of 0.94. The attention about a movie in the last seven days is calculated for predicting the box office revenue of a movie in a particular weekend. However, the box office revenue cannot be predicted based on only the amount of receiving attention.

Oghina et al. predict IMDb movie ratings by analysing twitter and YouTube data [10]. The model hypothesizes that correlations can be found between ratings provided by the Internet Movie Database (IMDb) and activity indicators around the same movie in other channels, such as social media. Two types of features named as surface and textual feature were extracted to build the model to predict the rating of a movie. Linear regression model was implemented in the WEKA toolkit to predict the rating of a movie. The research built several models based on different combinations of features. First linear regression models were built only with surface features extracted from twitter and YouTube such as likes, dislikes, views etc. Next, build models with a combination of surface and textual features. The dataset consisted of 70 movies, and their ratings as reported on IMDb on April 4, 2011 were used for experimentation. The data set was complemented with Twitter and YouTube data.

The performance of the models was measured using spearman's coefficient, ρ. Among the models built with only surface features, the model built with likes over dislikes gave the best results with a value of ρ equal to 0.4831. Then surface features were used to build models. The model built with textual features from twitter gave even better results with ρ equal to 0.7870. Textual features from YouTube showed a poor performance with ρ equal to 0.2768. Combined textual features from twitter and YouTube showed a better performance with ρ equal to 0.6625. However, the best result was found when a surface feature and textual features were used together to build a model to predict a movie's rating at IMDb. Its spearman's coefficient was 0.8539.

Different approaches of sentiment analysis and making predictions from twitter data were discussed. While all of them provide satisfactory results, it is still to be tested which approach performs the best when the classifiers are trained on the same dataset.

3. Proposed Methodology

An approach to classify tweets into positive and negative classes according to the sentiment expressed in a tweet is described in this section. The proposed framework is divided into following four steps and depicted with Fig. 1.

- Step-1: Data Input
- Step-2: Data Pre-processing
- Step-3: Feature Extraction
- Step-4: Tweet Classification Algorithms

These steps will be discussed in the following subsections.

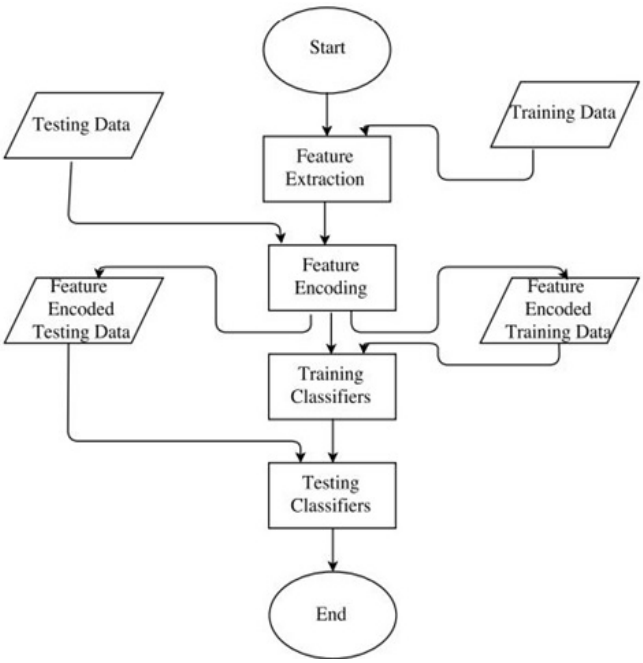

Fig.1. Classification Flow diagram

3.1. Data Input

Data input layer of this methodology deals with collecting twitter data and compiling training and testing sets. A pre-annotated dataset is collected and divided into a number of training and testing sets to be used as input in the model.

3.2. Data Pre-processing

Before extracting features from the tweeter dataset, the tweets have to be pre-processed in order to remove unwanted characters. This involves removing the user names, URLs, punctuation, and stop-words from the tweets. Then, shortened negations are elaborated such as 'don't' is converted to 'do not', 'havent' is converted to 'have not', etc. This process is repeated for every tweet in the dataset.

3.3. Feature Extraction

The textual data is not computable and thus not understandable by machine learning models. Therefore, feature extraction technique is used to convert text data into numerical representation. Four feature extraction techniques named as n-gram, lexicon, emoticon and intensifiers are applied for this purpose. The details of feature extraction strategies are discussed in the following sub-sections.

A. N-gram features

An n-gram is a contiguous sequence of n-items from a given sequence of text or speech. In twitter sentiment analysis, a set of *n* words is selected as features based on their likelihood in each document. The terms "so sad", "miss my", "so sorry", etc. are examples of bigrams. The n-grams were chosen using the following two approaches.

i) Categorical Proportional Difference (CPD): This is a feature selection technique for text categorization. CPD measures the degree to which a word contributes to differentiating a particular category from other categories [11]. In this method, a CPD value is assigned to every n-gram and n-gram with the highest score is selected as equation (1).

$$CPD(w,c) = \frac{A-B}{A+B} \tag{1}$$

Here, *w* stands for an n-gram and *c* stands for a class. *A* is the number of times word *w* and category *c* occur together, *B* is the number of times word *w* occurs without category *c*. CPD value for each n-gram for both positive and negative classes were calculated and then the maximum value was assigned to the n-gram.

ii) Chi Squared Value (Chi-square): Two events independency is measured through Chi-square test in statistics [12]. This technique is used in feature selection to test whether the occurrence of a specific term and class are independent, the calculation system is defined as equation (2).

$$x^2(w,c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{2}$$

Here, for an n-gram *w* and class *c*, *A* is the number of times word *w* and category *c* occur together. *B* is the number of times word *w* occurs without category *c*. *C* is the number of times category *c* occurs without word *w*. *D* is the number of times neither word *w* nor category *c* occurs and *N* is the total number of documents ($N = A + B + C + D$).

B. Lexicon Features

The lexicon is a dictionary or the vocabulary of a language. This approach relies on a lexicon or dictionary of words with pre-calculated polarity [13]. The pre-calculated polarity is used to score a document by aggregating the polarity of all the words in the document. The sentiment lexicon is an essential instrument in the field of tweeter classification [14].

If a word appears in a text, it will be compared with a word in the dictionary, and the sentiment score will be added to the total sentiment score of the text. For example, the total scores of the text: "I really enjoyed (+1) the webinar, it was fun! (+1): score = +2. I didn't like (-1) the webinar, because I hate (-1) the speaker: score = -2. The first case receives a sentiment score of +2, while the second case has a score of -2. The words contributing +/-1 towards the total score in each case are shown in brackets.

C. Emoticon Features

The term emoticon comes from "emotion and icon" and used to represent facial expressions by the sequence of text symbols [15]. These are used in the social media platform to convey emotional state with less writing. Looking side-ways the combination of characters becomes a happy and sad face which can replace an entire sentence. Emoticons

dominate text to convey sentiment in a sentence. Those are used in text to replace normal adjectives like happy, strong, sad, etc. For example, :-) and :-(represents a happy and sad face respectively.

D. Intensifiers

Intensifiers enhance the emotional context of the word by increasing the degree of the adjective or adverbs [16]. These adjectives or adverbs may be positive or negative sentiment words. The words 'strongly', 'pretty', 'incredibly', 'very', 'fairly' are examples of intensifiers. While performing sentiment analysis, these intensifiers need to be considered for obtaining more accurate results. It has been reported that a sentiment analysis method can be improved considering intensifiers [17].

3.4. Tweet Classification Algorithms

Sentiment analysis with machine learning algorithms has been proposed to classify tweets as positive or negative. For this study, Adaboost and Support Vector Machine were used as classifiers. The details regarding the algorithms are discussed below.

A. Adaboost

Boosting is an extremely powerful machine learning algorithm for text classification. Multiple poorly performing classifiers are combined to build a strong classifier and thus get high accuracy. This study focused on a well-known Adaboost algorithm which is a meta-estimator classifier [18]. The algorithm sets the classifier weight and trained the data sample to ensure the accurate predictions of unusual observations iteratively. Decision tree is used as a base learning classifier. One decision tree was found in single iteration with a specific weight based on the accuracy of the decision tree. Finally, all the decisions of these hypotheses are combined together with the following equation (3).

$$f(x) = \sum_{t=1}^{T} a_t h_t(x) \tag{3}$$

Here, T is the number of hypotheses; t is the index of the hypothesis h_t with weight a_t. Then, based on the sign of $f(x)$ the sample is classified into a class.

B. Support Vector Machine

Support vector machine is a discriminative machine learning classifier that learns by example to assign labels to objects [19]. The hyperplane can separate two classes appropriately and the training examples which are closest to the hyperplane are called support vectors [20]. The decision rule for support vector machine is represented as equation (4).

$$\sum a_i y_i \bar{x}_i \bar{u} + b \geq 0 \tag{4}$$

Here, $\bar{x}_i$ is a support vector, α_i is the language multiplier for $\bar{x}_i$ and y_i is 1 when $\bar{x}_i$ is positive and -1 when $\bar{x}_i$ is negative. $\bar{u}$ is the new observation that is needed to classify. This equation is used when the training set is linearly separable. In the above expression, the last part, $\bar{x}_i \bar{u}$ is the kernel function known as linear kernel. So, the decision rule can be rewritten as:

$$\sum a_i y_i K(\bar{x}_i \bar{u}) + b \geq 0 \tag{5}$$

If the output of the equation is greater than or equal to zero, it results as positive class, else negative class. K is the kernel function and the linear kernel is defined as equation (6).

$$K(\bar{x}_i, \bar{u}) = \bar{x}_i . \bar{u} \tag{6}$$

The decision rule for all variations of support vector machine is the same. Only the kernel functions differ. In this approach three different classifiers were built with three different kernel functions of SVM that are discussed below.

Radial Basis Kernel: This kernel function is used to find a non-linear classifier or regression line which is defined mathematically as equation (7).

$$K(\bar{x}_i, \bar{u}) = e^{(-\gamma |x_i|^2)} \tag{7}$$

Here, γ = *1/Number of features*.

Sigmoid Kernel: The sigmoid kernel is also known as hyperbolic tangent kernel which comes from the neural network field. This function is equivalent to a two-layer, perceptron neural network. This kernel function is represented as equation (8).

$$K(\bar{x}_i, \bar{u}) = \tanh(\gamma u x_i) \tag{8}$$

Here, γ = *1/Number of features*.

Polynomial Kernel: This is a non-stationary kernel that represents the similarity of vectors in training dataset in a feature space over polynomials of the original variables used. The kernel is mathematically represented as equation (9).

$$K(\bar{x}_i, \bar{u}) = (\gamma u x_i)^{\deg ree} \tag{9}$$

Here, γ = *1/number of features* and *degree* = 3.

The feature selection and tweet classification techniques used in this experimentation are discussed in this section. Two significant n-gram feature extraction schemes named as CPD and Chi-squared value were considered. On the other hand, the Adaboost and SVM with different kernel functions are applied for tweet classification purposes.

4. Experimentation & Result Analysis

The experimental dataset, experimentation and result analysis for sentiment analysis of twitter data is discussed in this section. The obtained results are compared to show effectiveness of different techniques. The accuracy and error rates of the classifiers are considered in classifying the test data set.

4.1. Environment Setup

The desktop workstation consisting of Intel core i5, 8GB RAM with Windows Operating System is used for this experimentation. The Visual Studio and RStudio were used for tweet processing and training the classifiers respectively.

4.2. Dataset

A pre-annotated dataset containing 1,578,627 classified tweets was collected from thinknook[2]. From this dataset, one training-set and three testing-sets were compiled. To this end, the tweets were chosen randomly from the dataset confirming uniqueness of each tweet. The dataset with a number of positive and negative tweets are shown in Table 1.

Table 1. Dataset

Dataset	Total Tweets	Positive Tweets	Negative Tweets
Training Set	62033	30000	32033
Test Set 1	8000	4061	3939
Test Set 2	4000	2000	2000
Test Set 3	4000	2000	2000

4.3. Experimentation

The feature extraction task was performed first then different classifiers were applied to classify the tweets as positive or negative.

Unigram and bigram features were extracted from both positive and negative tweets of the training dataset. Negation detection was applied during unigram and bigram extraction. If “no” or “not” preceded or conceded any of the words, those were added to the n-gram. So, if the tweet is “I do not like fish”, the extracted unigrams were “I”, “do+not”, “not+like”, “fish”. The extracted bigrams were “I do”, ”do+not like”, ”not+like fish”.

In this way, all the unigrams and bigrams present in the training set were extracted. The Table 2 shows sample n-gram features with CPD value. The n-grams which contribute to a sentiment classification have greater CPD value. Where, “very sad”, “really wanted”, “pleasure”, “poor thing” has very high CPD values because the n-grams contribute to differentiating sentiments. However, “copy”, “keyboard”, “cousins” have very low CPD value because the words do not contribute to any sentiment.

[2]http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-2012-09-22/

Table 2. N-gram CPD values

Sl.	N-gram	CPD Value
1.	very sad	1
2.	really wanted	0.93
3.	Pleasure	0.87
4.	poor thing	0.87
5.	Copy	0.04
6.	Keyboard	0.09
7.	Cousins	0.09

In Table 3, it can be seen that n-grams “sad”, “miss” and “good” have high value and contribute to differentiating positive and negative sentiment. But “monday”, “officially” and “think need” have very low Chi-square value because the words do not contribute to differentiating any sentiment.

Table 3. N-gram Chi squared

Sl.	N-gram	Chi-square Value
1.	Sad	862.582
2.	Miss	711.546
3.	Good	624.074
4.	good morning	124.831
5.	Monday	0.937
6.	Officially	0.127
7.	think need	0.61

Each tweet was assigned a prior polarity score based on the words present in them using MPQA subjectivity lexicon[3]. It is a list of 8000 common words with their prior polarity positive or negative. Two quantitative features named PriorPositive and PriorNegative were extracted based on the presence of any words from the lexicon in a tweet. For each positive or negative word, present in a tweet, PriorPositive or PriorNegative score is increased by 1. Negation detection was also performed while increasing the score. If a polar word was preceded by a “no” or “not”, the polarity score of the opposite feature was increased by 1.

A list of positive and negative emoticon features was compiled based on their usage in expressing positive and negative sentiment. Table 4 shows the sample list of positive and negative emotions.

Table 4. Positive and Negative Emoticons

Positive emoticons	:), :d, ;), :-), :p, =), (:,;-), xd, =d, ;d
Negative emoticons	:(, :/, :-(, =/, =(

Two quantitative features, PositiveEmo and NegativeEmo were extracted based on the number of positive or negative features present in a tweet.

Binary features were created based on the presence of intensifiers. For tweet analysis, all-caps words and words with character repetitions are considered as intensifiers such as “LOVE”, “hellllloo”, etc.

Finally, the tweets were encoded with the extracted features for training and testing dataset. The encoded values were written in a file as a matrix. Each row of the matrix represented a tweet and each column except the last column represented a feature.

A. Classification with Adaboost

Adaboost models were built with decision trees as weak learners and iterated 10 times. All the features except n-grams are common in both classifiers for n-grams with Chi-square and CPD value. The performance of the adaboost classifiers for n-gram selected with Chi-square (Chi2) and CPD is shown in Table 5. The precision and accuracy of this model is higher for Chi-square than CPD. Where, the highest recall value is 0.85 which is achieved by CPD.

[3]http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

Table 5. Adaboost Classifier Performance

DS	Precision		Recall		Accuracy	
	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.67	0.63	0.80	0.85	69.5	66.6
TS2	0.68	0.63	0.81	0.85	71.4	67.6
TS3	0.66	0.62	0.80	0.85	69.7	66.6
Avg	0.67	0.63	0.80	0.85	70.2	66.9

B. Classification with SVM

SVM classifiers were built using three different kernel functions for both Chi-square and CPD features which are discussed below.

a. SVM Radial Basis Kernel:

The classification performance of SVM radial basis kernel for n-grams selected with Chi-square and CPD is illustrated in Table 6. The highest precision and accuracy is 0.72 and 74.3 respectively which are measured under Chi-square. On the other hand, CPD scored highest recall which is 0.89. All the features except n-grams are common in both classifiers trained for n-grams selected with Chi-square and CPD value.

Table 6. SVM with Radial Basis Kernel

Dataset	Precision		Recall		Accuracy	
	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.70	0.65	0.81	0.86	72.7	69.6
TS2	0.72	0.66	0.81	0.87	74.2	70.5
TS3	0.71	0.66	0.83	0.89	74.3	71.2
Avg	0.71	0.65	0.81	0.87	73.7	70.4

b. SVM Sigmoid Kernel:

The tweet classification performance of SVM sigmoid kernel for n-grams selected with Chi-square and CPD value is shown in Table 7. Where, the precision and accuracy is greater for Chi-square and recall is higher for CPD. The features except n-grams are common for both Chi-square and CPD values.

Table 7. SVM with sigmoid Kernel

Dataset	Precision		Recall		Accuracy	
	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.70	0.66	0.82	0.87	73.14	70.1
TS2	0.72	0.66	0.83	0.87	74.97	70.85
TS3	0.71	0.66	0.85	0.89	75.1	71.55
Avg	0.71	0.66	0.83	0.88	74.40	70.83

c. SVM Polynomial Kernel:

The SVM polynomial kernel classifier performance is reported in Table 8. Where the precision and accuracy is always higher for Chi-square and recall is higher for CPD. The results are shown for n-gram features selected with Chi-square and CPD value. Only n-gram features were different in this case for both Chi-square and CPD consideration.

Table 8. SVM with Polynomial Kernel

Dataset	Precision		Recall		Accuracy	
	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.63	0.61	0.88	0.90	67.98	65.67
TS2	0.64	0.61	0.88	0.91	68.82	66.22
TS3	0.63	0.61	0.91	0.93	69.22	66.47
Avg	0.63	0.61	0.89	0.91	68.67	66.13

4.4. Result Analysis

The tweet classification efficiency for proposed models named as Adaboost, SVM radial basis kernel, SVM sigmoid kernel and SVM polynomial kernel is demonstrated in the following Tables. Where, Table 9 showed the

classification precision for both feature selection methods i.e. Chi-square and CPD value. N-gram is considered for both Chi-square and CPD techniques.

In Table 9, it can be seen that the average precision of the classifiers trained with n-gram features selected with Chi-square value is always greater than the precision of CPD value. As stated in Table 9., the highest value is 0.72 which is achieved by SVM radial basis and sigmoid under Chi-square technique. Where the lowest precision is 0.61 and falls under SVM polynomial for CPD.

Table 9. Precisions of different Methods

Data Set	Adaboost		SVM Radial Basis		SVM Sigmoid		SVM Polynomial	
	Chi2	CPD	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.67	0.63	0.70	0.65	0.70	0.66	0.63	0.61
TS2	0.68	0.63	0.72	0.66	0.72	0.66	0.64	0.61
TS3	0.66	0.62	0.71	0.66	0.71	0.66	0.63	0.61
Avg	0.67	0.63	0.71	0.66	0.71	0.66	0.63	0.61

On the other hand, the recall for experimented classifiers is shown in Table 10. The achieved recall of the classifiers with n-gram for both Chi-square and CPD value is reported. The average recall of the classifiers for CPD is always greater than Chi-square. Where the highest and lowest recall is 0.93 and 0.80 respectively.

Table 10. Recalls of different methods

Data Set	Adaboost		SVM Radial Basis		SVM Sigmoid		SVM Polynomial	
	Chi2	CPD	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	0.80	0.85	0.81	0.86	0.82	0.87	0.88	0.90
TS2	0.81	0.85	0.81	0.87	0.83	0.87	0.88	0.91
TS3	0.80	0.85	0.83	0.89	0.85	0.89	0.91	0.93
Avg	0.80	0.85	0.82	0.87	0.83	0.88	0.89	0.91

The accuracy of Adaboost and three SVM kernel models is reported for different datasets in Table 11. Where, the accuracy is always better for Chi-square compared to CPD strategy.

Table 11. Accuracy comparison

Data Set	Adaboost		SVM Radial Basis		SVM Sigmoid		SVM Polynomial	
	Chi2	CPD	Chi2	CPD	Chi2	CPD	Chi2	CPD
TS1	69.5	66.6	72.7	69.6	73.1	70.1	68.0	65.7
TS2	71.4	67.6	74.2	70.5	75.0	70.9	68.8	66.2
TS3	69.7	66.6	74.3	71.2	75.1	71.6	69.2	66.5
Avg	70.2	66.9	73.7	70.4	74.4	70.8	68.7	66.1

The accuracy of experimented classifiers for Chi-square and CPD is demonstrated in Table 11 and plotted in Fig.2. As stated in the table average accuracy of Adaboost for Chi-square and CPD were 70.2% and 66.9% respectively. Where, the average accuracy of SVM radial basis kernel and polynomial kernel with Chi-square n-grams were 73.73% and 68.67% respectively. On the contrary, the average accuracy of the SVM sigmoid kernel with Chi-square n-grams is 74.4%. So, this experiment shows that the sigmoid kernel of SVM fits the dataset and best followed by radial basis kernel and polynomial kernel.

The Fig. 2 depicted the accuracy comparison of investigated results, where accuracy percentage is denoted vertically (x-axis). The Adaboost, SVM radial basis, SVM sigmoid and SVM polynomial classifiers with Chi-square and CPD are represented horizontally. The first, second, third and fourth bars represent results for test dataset-1 (TS1), dataset-2 (TS-2), dataset-3 (TS-3) and average (avg) respectively.

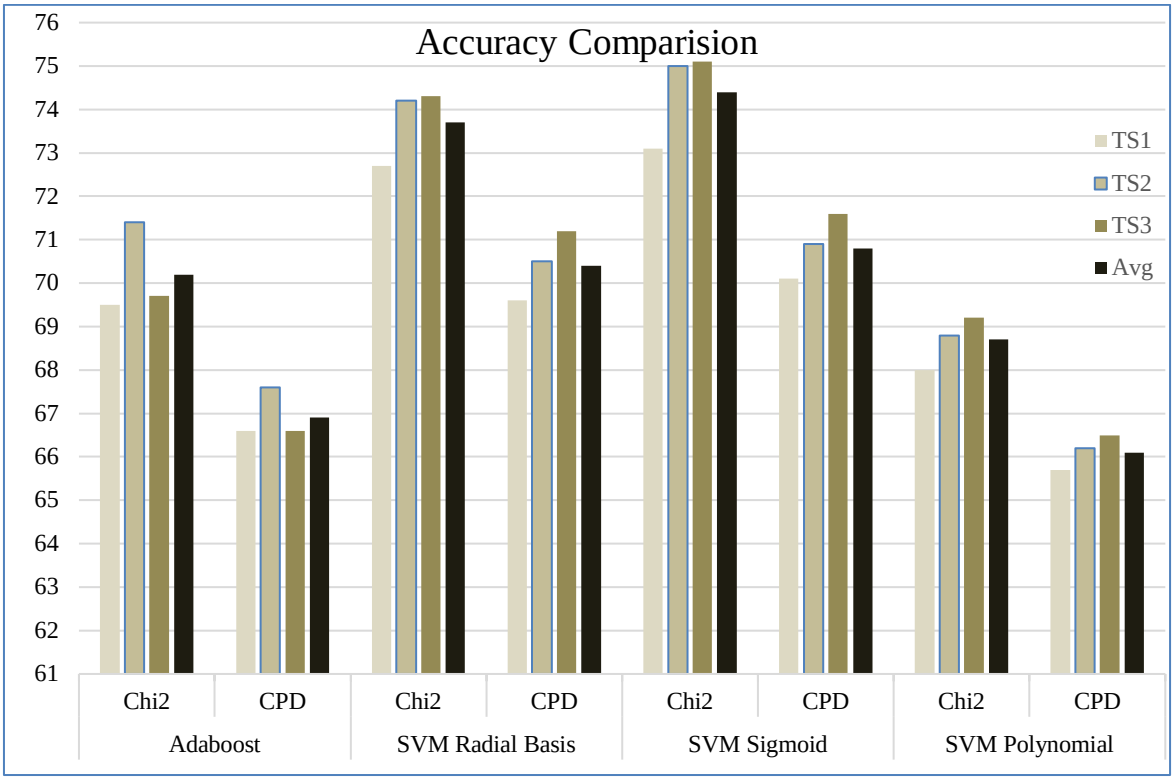

Fig.2. Accuracy of Adaboost and SVM kernels

A. Discussion

According to Table 9, 10 and 11, it is found that precision and accuracy is higher for Chi-square than CPD. So, it can be concluded that n-grams selection with Chi-square is a better feature selection technique than n-grams selection with CPD value. The reason behind is that, CPD value does not consider the number of observations; a feature is not present in. The ratio between the differences of number of observations; A feature is present in different classes to the sum of the observations a feature is present in both classes. For 1000 positive and negative tweets, if the n-gram "x" is present in only one positive tweet and not in any negative tweets, the CPD value will be as below.

$$CPD(x, positive) = \frac{1-0}{1+0} = 1$$

Here, the n-gram "x" occurs only once in the training set, still its CPD value is 1 which is the highest range of CPD value. But it is clear that the n-gram "x" is not significant in differentiating the classes as it occurs only once. On the other hand, if an n-gram "y" occurs 500 times in positive class and 10 times in negative class, its CPD value for positive class will be as below.

$$CPD(y, positive) = \frac{500-10}{500+10} = 0.96$$

The n-gram "y" was more significant in differentiating positive and negative classes but n-gram "x" has a higher CPD value. So, n-gram "x" will be picked as a feature over "y" for the CPD feature selection system. However, if Chi-square is measured for both "x" and "y" in positive class, the Chi-square value of "y" is greater than "x" as below. Because, Chi-square considers the number of observations in which a feature does not occur.

$$X^2(x, positive) = \left(\frac{2000*(1000-0)^2}{(1+999)(0+999)(1+0)(999+1000)} \right) = 1.001$$

$$X^2(y, positive) = \left(\frac{2000*(495000-5000)^2}{(500+500)(990+10)(500+10)(500+990)} \right) = 631.925$$

Therefore, “y” will get picked over “x”. Therefore, the more significant feature is selected. That is why feature selection with Chi-square performs better.

On the other hand, the average accuracy of the classifier trained with SVM sigmoid kernel and Chi square n-grams has the best accuracy of 74.4% and has a precision of 0.711. It outperforms all other variations of the SVM. The average accuracy of SVM radial basis kernel and polynomial kernel with Chi-square n-grams were 73.73% and 68.67% respectively. The accuracy of the adaboost classifier was 70.19% which is lower than SVM sigmoid and SVM radial basis classifiers. Although adaboost usually performs better than SVM, it can be more susceptible to overfitting the data. Since, here using a very large number of features, adaboost classifiers are more likely to overfit in this case.

On the contrary, SVM maximizes the distance between the observations of two classes. In this case, SVM classifier creates a hyperplane which maximizes the distance between positive and negative tweets. So, it is less likely to overfit the training data. That is why, SVM with radial basis kernel and SVM with sigmoid kernel are performing better than adaboost. However, adaboost classifier performed better than the SVM with polynomial kernel because the biggest disadvantage of SVM lies in choice of the kernel [21]. If a kernel is chosen that cannot create a hyperplane which can separate the classes with a satisfactory margin, the performance of the classifiers will be bad. From the experiments, it is clear that SVM polynomial kernel is not a very good choice for sentiment analysis because it cannot create a hyperplane which can separate the classes with a satisfactory margin. Thus leads to a poor performance of the classifier.

In a nutshell, the experimentations for classifying the tweets were conducted with Adaboost and three SVM kernels. The features used for training the models were n-gram presence, positive emoticon presence, negative emoticon presence, prior positive score, prior negative score and presence of intensifiers. Analyzing the demonstrated results, it is concluded that SVM Sigmoid kernel for n-grams selected with Chi-square value performed the best.

5. Conclusions

This research investigated different approaches in selecting features for performing sentiment analysis in twitter data. Performance of different machine learning algorithms in training classifiers for sentiment analysis was evaluated. Where, Chi-square shows higher precision and accuracy compared to CPD. The average accuracy of Adaboost for Chi-square and CPD were 70.2% and 66.9% respectively. On the other hand, the average accuracy of SVM radial basis kernel and polynomial kernel with Chi-square n-grams were 73.73% and 68.67% respectively. Where, the average accuracy of the SVM sigmoid kernel with Chi-square n-grams is 74.4%. The best accuracy found in this experiment for twitter sentiment analysis was 74.4% for SVM with sigmoid kernel using Chi-square n-grams selection. Investigating the demonstrated results, it is concluded that n-grams selection with Chi-square is a better feature selection choice than n-grams selection with CPD value.

While 74.4% is a good accuracy rate, there is still room for improvement. There are many other machine learning algorithms and feature selection methods still to be explored. This twitter sentiment analysis work will be expanded by implementing SVM and Adaboost with other feature extraction techniques e.g. BoW, TF-IDF.

References

[1] Shaheen M, Awan SM, Hussain N, Gondal ZA. Sentiment Analysis on Mobile Phone Reviews Using Supervised Learning Techniques. International Journal of Modern Education and Computer Science. 2019 Jul 1;11(7).

[2] Yasin Görmez, Yunus E. Işık, Mustafa Temiz, Zafer Aydın, "FBSEM: A Novel Feature-Based Stacked Ensemble Method for Sentiment Analysis’ Comments in E-Government", International Journal of Information Technology and Computer Science, Vol.12, No.6, pp.11-22, 2020.

[3] Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." LREc. Vol. 10. 2010.

[4] Wang, Hao, et al. "A system for real-time twitter sentiment analysis of 2012 us presidential election cycle." Proceedings of the ACL 2012 System Demonstrations. Association for Computational Linguistics, 2012.

[5] Kouloumpis, Efthymios, Theresa Wilson, and Johanna D. Moore. "Twitter sentiment analysis: The good the bad and the omg!." Icwsm 11 (2011): 538-541.

[6] Silva, Nádia Félix Felipe da, Eduardo Raul Hruschka, and Estevam Rafael Hruschka Junior. "Biocom_Usp: tweet sentiment analysis with adaptive boosting ensemble." International Workshop on Semantic Evaluation, 8th. ACL Special Interest Group on the Lexicon-SIGLEX, 2014.

[7] Barbosa, Luciano, and Junlan Feng. "Robust sentiment detection on twitter from biased and noisy data." Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics, 2010.

[8] Hossam Elzayady, Khaled M. Badran, Gouda I. Salama, "Arabic Opinion Mining Using Combined CNN - LSTM Models", International Journal of Intelligent Systems and Applications, Vol.12, No.4, pp.25-36, 2020.

[9] Asur, Sitaram, and Bernardo A. Huberman. "Predicting the future with social media." Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on. Vol. 1. IEEE, 2010.

[10] Oghina A, Breuss M, Tsagkias M, De Rijke M. Predicting imdb movie ratings using social media. InEuropean Conference on Information Retrieval 2012 Apr 1 (pp. 503-507). Springer, Berlin, Heidelberg.

[11] Simeon M, Hilderman R. Categorical proportional difference: A feature selection method for text categorization. InProceedings of the 7th Australasian Data Mining Conference-Volume 87 2008 Nov 27 (pp. 201-208).

[12] Yang Y, Pedersen JO. A comparative study on feature selection in text categorization. InIcml 1997 Jul 8 (Vol. 97, No. 412-420, p. 35).
[13] Kolchyna O, Souza TT, Treleaven12 PC, Aste12 T. Methodology for twitter sentiment analysis. arXiv preprint arXiv:1507.00955. 2015 Jul.
[14] Alsolamy AA, Siddiqui MA, Khan IH. A Corpus Based Approach to Build Arabic Sentiment Lexicon. International Journal of Information Engineering and Electronic Business. 2019 Nov 1;11(6).
[15] Hallsmar F, Palm J. Multi-class sentiment classification on twitter using an emoji training heuristic.
[16] Mukhtar N, Khan MA, Chiragh N, Nazir S. Identification and handling of intensifiers for enhancing accuracy of Urdu sentiment analysis. Expert Systems. 2018;35:e12317.
[17] Strohm, F., 2017. The Impact of intensifiers, diminishers and negations on emotion expressions (Bachelor's thesis).
[18] Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences. 1997 Aug 1;55(1):119-39.
[19] Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. InProceedings of the fifth annual workshop on Computational learning theory 1992 Jul 1 (pp. 144-152).
[20] Bahar Nazlı, Yasemin Gültepe, Hayriye Altural. " Classification of Coronary Artery Disease Using Different Machine Learning Algorithms ", International Journal of Education and Management Engineering, Vol.10, No.4, pp.1-7, 2020.
[21] Prabhu, N. "Gauge groups and data classification." Applied mathematics and computation 138.2 (2003): 267-289.

Authors' Profiles

Md. Abdur Rahman received his BSc in Information Technology from Visva Bharati University, India in 2004. He has completed his Post Graduate Diploma and Master in Information Technology from University of Dhaka, Bangladesh, in 2008 and 2009 respectively. He is a Senior Computer Scientist in the Centre for Advanced Research in Sciences at the University of Dhaka. His major research interest includes text analytics, application of machine and deep learning in software engineering, and natural language processing. He has published a number of research papers in various international journals and conferences.

Mobashir Sadat graduated with a BSc in Software Engineering from University of Dhaka. His research interests lie in the areas of natural language processing, machine learning and information retrieval. Specifically, he is interested in sentiment analysis, text summarization, natural language inference and hierarchical topic classification. Currently, he is a computer science PhD student at University of Illinois at Chicago.

Saeed Siddik has been working on Software Testing and Software Analysis research where he experimented how software are developed and tested efficiently. He has completed his M.Sc. in Software Engineering, including the highest marked thesis dissertation on Software Test Case Prioritization from IIT University of Dhaka. The research outcomes of that thesis were published at several Journal and Conferences. He was the first research student of IITDU Optimization Research group, where he was working on software design migration to enhance modularity and manageability. He is a member of IEEE (ID:94159542).

How to cite this paper: Abdur Rahman, Mobashir Sadat, Saeed Siddik, "Sentiment Analysis on Twitter Data: Comparative Study on Different Approaches", International Journal of Intelligent Systems and Applications(IJISA), Vol.13, No.4, pp.1-13, 2021. DOI: 10.5815/ijisa.2021.04.01