DOI: 10.14421/ijid.2019.08206 IJID (International Journal on Informatics for Development), *e-ISSN:2549-7448*
Vol. 9, No. 2, 2019, Pp. 84-94

Sentiment Analysis using Twitter Dataset

Ibrahim Moge Noor[1], Metin TURAN[2]
Department of Engineering, Faculty of Computer Engineering
Istanbul Commerce University
Istanbul, Turkey
[1]engineermoge@gmail.com, [2]mturan@ticaret.edu.tr

Article History
Received Feb 9th, 2020
Revised Aug 25th, 2020
Accepted Aug 25th, 2020
Published Dec, 2019

***Abstract*— Apparently Social media sites are becoming increasingly popular, it creates platforms through which organizations, communities, and individuals share and discuss various topics. The reviews and data obtained from these sites are essential for further analysis. In this paper we studied the sentiment classification of 2019 Kenyan 1000 banknote demonetization using Twitter as our source dataset. We perform Multi nominal naïve Bayes classifier algorithm to classify tweets documents. We split our dataset using k-folder validation since we had limited amounts of data, so to achieve unbiased prediction of the model. . We obtained in test data an accuracy of 70.8% when we used unigram model and 64.1% when we applied bigram model. Results show that the model reached to an acceptable accuracy of (71%) on average using unigram model.**

***Keywords*— Machine Learning; Multinomial Naïve Bayes; Sentiment Analysis; Twitter Data; N-gram**

1 INTRODUCTION

Nowadays social media platform became popular with billions of users around world, sharing their opinions of a particular subject, as more Internet users share their opinion daily it becomes a valuable source of data. Sentiment analysis techniques is good ways to identify and find opinions of the authors by expressing into polarity positive, negative or neutral [1].

Demonetization is withdrawal of currency from circulation and replace the old currency to new currency [2] After demonetization in Kenya, the citizen across nation post their view on demonetization via social media. Explicitly we use Twitter as a source of our data collection.

Twitter [3] is one of the best online social network site, the microblogging service which has also become an important source of real-time events updates, and had over 48.6 million active users and 330 million active users per month, where it plays an important role in expressing our feelings [4] where users can share either opinions or information about product, events or politics. Each tweet is restricted to a limit of characters where user can post a short message [5].

1.1 Tweet feature

- Length of a tweet: the maximum characters per tweet is 280 characters, even though some user use abbreviation like 'b4','ur','u8','g8t','sry','coz','pic' which is not meaningful and grammatically correct sentence, it can be consider as sentence.
- Language: people use Twitter in various languages, though we just considered English language tweets beside Kenya being multilingual country, 90% of Twitter user tweet with English language.
- Hashtag: are formed by using the pound sign **(#)** in front of the word with no space and punctuation like #Kenyanewcurrency, it make conversation cantered around the same topic easier to search, this help us when we extracting tweets from Twitter.

Web scraper with help of Twitter advance search was used to extract tweets data from Twitter [6], On related topic as discussed above.

1.2 Classification of Sentiment Analysis

Our sentiment index relies critically on tracking the reference frequencies of Vocabularies with positive and negative connotations [7]. The extraction of the sentiment can be in several level the most common are phrase level, sentiment level extraction of sentiment are done from each sentence [8], [9].

Sentiment analysis was considered a grouping issue. Much the same as in enormous reports, sentiments of tweets can be communicated in various manners and characterized by the presence of sentiment, i.e., if there is sentiment in the tweets, contain polar words then it is assign either positive or negative, else it is viewed as Neutral. As there are words in the text of the both two classes, they don't give any significant data. The studies shows that to applied term frequency inverse document frequency (TF-IDF) metric in order to solve this kind of problem [10].

Some authors they categorize sentiment of text into six emotions sadness, anger, disgust, fear, joy and surprise [11]. In order to classify the sentiment behind the tweets, count the negative and positive words allocate a score for each tweet. In view of the score, the tweet will be classified into negative, positive and neutral. Extremity scores are additionally relegated to each tweet based emotional of tweets such joy, sad, happiness or anger likewise, and base on polarity such negative, positive and neutral.

The supervised learning techniques need corpus of which was classified before into specifics grouped so that can be used in machine learning purposes, These pre-grouped datasets are regularly domain, the model it create can work just for a specific domains. These datasets are first changed over into transitional models where records are spoken to as vectors, so that these converted data can be used to feed machine learning algorithm [12].

Tweets post are unlike the other social media sites, they are short and normally show limited sentiment signals. Unbiased tweets are substantially more average than negative and positive tweets. Which will as a rule be overwhelmingly positive or negative.

Sentiment Analysis inside a multilingual tweets offers various difficulties. Statistical methodologies require training material which is ordinarily sparse for various dialects. Then again, lexical methodologies require language explicit lexical and semantic assets. Creating these assets is very tedious and requires regularly manual work. As per our knowledge, there are chiefly two approaches that are important with regards to multilingual sentiment analysis. A corpus based approach and a dictionary based way to deal with multilingual subjectivity analysis (abstract versus objective). Inside the dictionary based methodology, an objective language subjectivity classifier is produced by interpreting a current dictionary. The corpus-based methodology constructs a subjectivity-commented on corpus for the objective language through projection. A factual classifier is prepared on the subsequent corpus [13].

Machine learning demands a huge dataset for training purpose in order to obtain high accuracy. The topic is new and no one collected data relating to subject before, tweets gathered and labelled manually by ourselves although it was time consuming for filtering and removing the noises.

1.3 Multilingual Tweets

The utilization of twitter as a social network in Kenya is at present developing a great deal of possibly helpful data is being gone through the network. This data can give some an incentive to researchers or scientists on the overall view of their items or services.

However, some Kenyan twitter users normally tweets with mixed language such as English and Swahili and this blend is normally unstructured and casual. However, most of collected data was in English language and the little remained was translated to English language.

The dataset collected was mixed with another languages beside English, mostly Swahili language, one of the tweet mixed with Swahili word is shown in Figure 1 so we had to translate this word to cross pounding English word.

Figure 1. Tweet mixed with mixed with Swahili language

We come across such abbreviation problem as shown in Figure 2, and the abbreviation were changed to its original words.

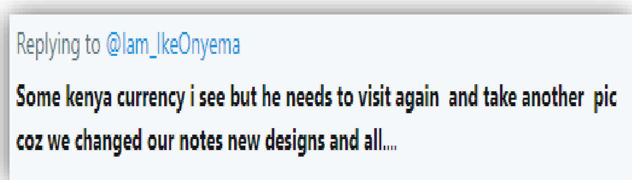

Figure 2. Abbreviation used in tweets

2 METHOD

Sentiment analysis is the automated process of understanding an opinion about a given subject from spoken language or written. Sentiment analysis became a trend topic of various researches and text mining has been done in past years. Multinomial Naïve Bayes method used specifically addressing recurrence in the content of the document. The Multinomial Naïve Bayes model has been introduced as an option of Naive Bayes for text classifier. In recent past years, many researchers usually regard it as the ideal Naive Bayes text classifier [14]. Multinomial Naive Bayes a family of probabilistic classifiers, the state of art of Bayesian classifier is the best since it is fast and simple text classifier [15]. TF-IDF substitution relatively improved the performance of the general classifier [16], [17]. TF-IDF measures word scores effectively before characterization. TF-IDF was straightforward, actualize and process. Multinomial Naive Bayes improved considerably by applying a TFIDF change to the word features as well as weight learning [18].

The supervise machine learning are tend to be more accurate since each of the classifiers is trained on an assortment of representative data called corpus however the supervise machine learning depends on the quality of training data as well the type of algorithm used [19]. The collected dataset from Twitter was labelled into different polarities positive, negative and neutral, labelling data is scarce and time consuming [20]. Then was classified to their respective class using machine learning algorithms with unigrams and bigrams as features [21]. The drop of accuracy in n gram for some text classification algorithm may cause by sparsity of data [22].

The polarity of tweets such positive and negative, neutral in tweets were studied [23]. One of the techniques that sentiment analysis can be conducted is lexicon-based approach in which, the dictionary is made out of a lot of positive and negative assessment words, used to score the tweets either, positive, negative or neutral [24].

Sentiment analysis techniques are good ways to identify and find opinions of the authors by expressing into polarity positive, negative or neutral [25]. implemented supervised algorithms, they compared different feature extraction determine which algorithm is best suited in term of execution time for Sentiment Analysis based on the given dataset [25].

2.1 Data Preprocessing

Data Pre-processing: is a procedure that is utilized to change over the crude information into a clean data set. The data we extracted from

Twitter site was in raw format which is not feasible for the analysis.

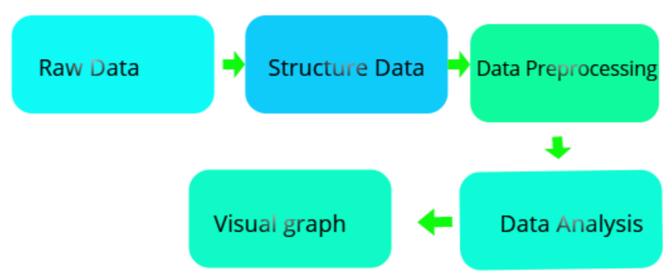

Figure 3. System Flowchart

Data pre-processing is one way of preparing the data in a way that is suitable to analyse as shown in Figure 3 above. 1087 tweets, was extracted from Twitter, between 1st June to 11th October, although there was some shortage of tweets, the dataset was collected using web Scraper and Twitter advance search. it was consist of 431 negative tweets, 332 neutral tweets, and 324 positive tweets. The collected data was split into test and train datasets: 967 tweets for training and 120 for the test.

Demonetization data was collected between June and October, then the gathered data was applied important techniques in order to reduce the noise and dimensionality of sentence. The data was cleaned by removing symbols, non-English words, extra spaces, and numbers. Also, the collected tweets were mixed of hashtags '#', url links, annotation '@', also we remove the stop words, these are common words that don't add value for classification such as and, either, to, the, so on. Stemming also was applied, we take out the root of the word.

2.2 *Feature Extraction*

The extracted tweets were stored in unstructured was stores as text format. This unstructured data supposed to change over meaningful data in order to feed it to a machine learning algorithm. The algorithm needs numerical vectors and not textual data, in order to convert text into corresponding integers, the vectorization of the text file to numerical vectors is done utilizing the following approaches.

2.2.1 *The Term Frequency–Inverse Document Frequency (TF–IDF)*

A numerical measurement that is aimed to reflect how important word is to corpus or a document, which is used in machine learning and text mining as a weighting plan in data recovery that has additionally discovered great use in archive characterization.

When weight increases as the word frequent in document increases but is offset by the frequency of the word in a document, the offset TF–IDF contains two elements term frequency and inverse document frequency, is calculated as follow:

- TF = (Frequency of a word in the document) / (Total of words in the document).
- IDF =Log ((Total number of documents) / (Number of documents containing the word)).

2.2.2 *Count Vectorization*

Count Vectorization gives a straightforward method to both tokenize collection of text documents and create a vocabulary of known words as well encode new document by utilizing that vocabulary, which will produce a sparse representation of counts. We use Count Vectorization in our dataset as follows. We created vectors that have a dimensionality equivalent to the size of our Sentiments which is either negative, positive or neutral, so if the content data features that sentiment word, we will put a one in that dimension and rest assign zero, each time we experience that word once more, we increased the count.

```
Train words

"Precious ","1","0","0"
"Debates","0","1","0"
"shame","0","1","0"
"abuse","0","1","0"
"treason","0","1","0"
"Could this","0","0","0"
```

Figure 4. Train Words

As illustrated in Figure 1 above, if the word is positive the first columns were assigned one, else if the word is negative the second columns were assigned one or else all column were assign zero for neutral words.

2.3 *N-gram mode*

Applying N-gram model in the sentiment analysis is very helpful in analyzing the sentiment of document or text.in this paper only unigram which refers to n-gram of size one and bigram which refers to n-gram of size two was applied. N gram is used for improving features for supervised machine such as Naïve Bayes. In our Dataset there was over 3000 vocabularies, most these vocabularies had low frequency so perform pruning in order to reduce over fitting and complexity of classifier also it improve our model accuracy. We have only use the most effective vocabularies, we train each vocabularies as its respective polarities positive, negative or neutral. We have created dataset with sentiment classification by preparing negative word corpus, words that is disagreeing with demonetization process, positive word, words that agreeing with demonetization events as well we created neutral words, word that neither agreeing nor disagreeing . Each tweet is assessed and a numeric score is calculated. In view of this score, the labels sentiment are connected by the accompanying rules. If positive score is more than negative score was assign as positive, else if negative score is more than positive score then was assign as negative. If both negative and positive score are equal then was assign as neutral. Some of the unigram vocabularies is shown in table 1 below.

Table 1.unigram words

Positive	**Negative**	**Neutral**
stash	crisis	launch
accessible	monopoly	visit
appreciate	difficult	release
innovate	claims	return

Bigrams, where tokens represents two consecutive vocabularies as shown in table 2 below, we have extracted the information gain and useful in our training model as well ignore the least ones.

Table 2.Bigram Word

Positive	Negative	Neutral
accessible public	fake money	caution aware
advocate less	foreign currency	enough aware
agree commitment	awkward realisation	Exchange ksh (Kenyan shilling)
curb fraud	flow integration	announces plan
flows counterfeit	flimsy excuses	caution public
tackle illicit	Felix discuss	laud launch
safety country	bear image	application helps
security features	court challenge	forward curb

2.4 Sentiment Analysis of Tweets

It's estimated that 80% of the world's information is unstructured and not sorted out in a pre-characterized way.

Sentiment analysis algorithms: There are various methods that can be used to implement sentiment analysis, see in Figure 5, which can be group as:

1. *The automatic system depends on machine learning techniques to learn the data.*
2. *The rule-based system which performs sentiment analysis by a set of physically created principles.*
3. *The hybrid system combines both.*

Machine learning and Rule-based system approach to address Sentiment Analysis are called Hybrid. We perform our Twitter sentiment analysis using the Multinomial Naive Bayes algorithm which is a type Naive Bayes, Naive Bayes classifier to get higher accuracy and we come up with a lexicon analysis that contains a word list which is negative and positive or neutral.

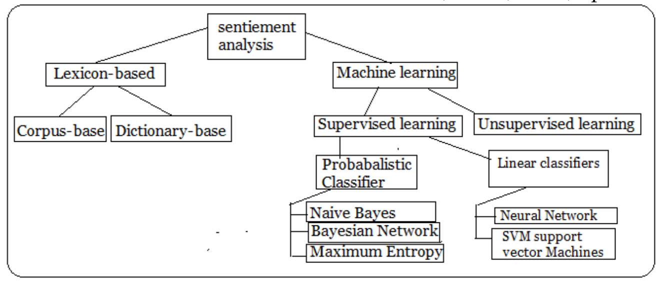

Figure 5. Sentiment analysis methods. Classification of Sentiment Analysis Researches.

In order to classify the sentiment behind the tweets, count the negative and positive words to allocate a score for each tweet. In view of the score, the tweet will be classified into negative, positive and neutral. Extremity scores are additionally relegated to each tweet based emotional of tweets such as joy, sadness, happiness or anger likewise, and base on polarity such negative, positive and neutral.

2.5 Multinomial Naive Bayes method

Naive Bayes is a classification method which is based on Bayes' theorem. It is suitable for large data sets since it assumes independence between predictors and it assumes that a feature in a class is which is not related to any other also is fast it only needs one pass over the data, Figure 6.

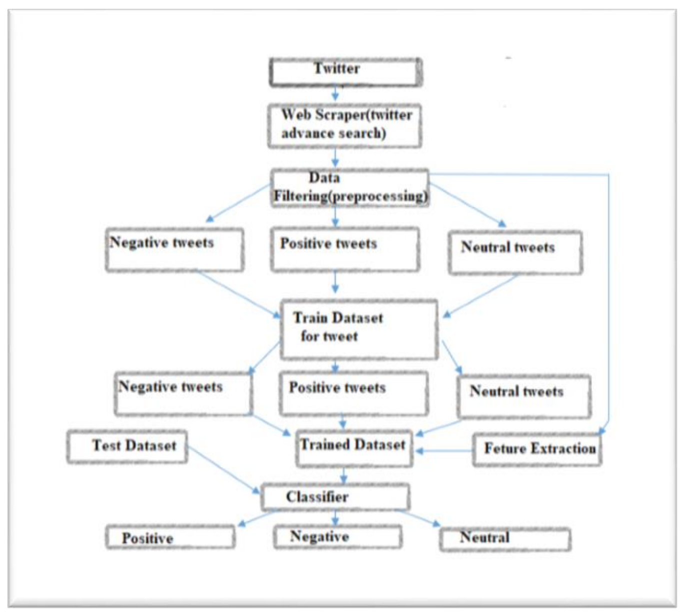

Figure 6. Sentiment Classification Based On Emoticons

We perform our sentiment classification with Multinomial Naïve Bayes which is a type of Naïve Bayes. Multinomial Naïve Bayes method used to represent the recurrence in the text of the document. It is a probabilistic classifier, and there are two fundamental methodologies you could take, to train a model in order to recognize the polarity tweets (positive, negative or neutral). A supervised which is an object of interest in this article and an unsupervised one. The first would ask from you to gather labelled data, and train the algorithm, in a supervised way how each word in a grouping relates to the result of in general sentence being positive, negative or neutral. This methodology requires physically marked data, which is regularly tedious, and not constantly conceivable. Unsupervised learning is that you don't give any past presumptions and definitions to the model about the result of factors you feed into it, you just supplement the information and need the model to become familiar with the structure of the data itself.

We found first the prior probability of our document by just dividing the number of documents of that class (either positive, negative and neutral) by the total number of the document.

$$\Pi(\chi) = \frac{Nc}{N} \tag{1}$$

We calculate for the word given in a class, P (w|c) = $\frac{Count\ (W,C\ )\quad +1}{Count\ (c)+\quad |V|}$, the Addison of 1 and v (vocabulary), for smoothing purpose in case some word got zero counts,

We have used the sum of logs to avoid underflow.

$$\Pr(c) \propto \prod_{w=1}^{|v|} \Pr\ (w|c)^{fw} \tag{2}$$

We have used the sum of logs to avoid underflow.

$$\Pi\rho\ (c)\ \alpha\ \lambda o\gamma\ (\pi c\angle\ \Pi\rho\ (w|c)\ fw|V|w{=}1) \tag{3}$$

2.6 Confusion matrix

We use a confusion matrix to summarize the performance and prediction results of our classification algorithm, Confusion matrix it give us a better idea of our classification model.

The dataset utilized for the experiments was divided into 3 classes, positive, negative and neutral (1, 2 and 3 respectively).

To validate our results, accuracy, precision, recall, and f1-score metrics are calculated.

Accuracy: Overall, how often is the classifier correct?

- Negative (N) - Observation is not positive
- Positive (P) - Observation is positive.
- (FP) False Positive - Observation is negative, but is predicted positive.
- (FN) False Negative (FN) - Observation is positive but is predicted as negative.
- (TP) True Positive - Observation is positive and predicted to be positive.
- (TN) True Negative - Observation is negative and is predicted to be negative.

Precision

- TP/total predicted positive.
- It is calculated as TP/ (TP+FP).

Recall

- it calculates how many of the Actual Positives our model capture through labelling it as Positive: using the following formula. TP / (TP+FN).
- F Score
- This is a weighted average of the true positive rate precision and recall.it is calculated as 2TP/ (2TP+FP+FN).

2.7 Data validation

Validation is a significant step that permits us to test the accuracy of our model. The most well-known ways to deal with validation are

- Hold out technique
- Cross validation strategy.

In the hold out strategy, part of the information is held out for testing and the rest of the dataset are utilized for training the classifier. The cross validation technique, by comparison, we split our dataset into testing and training, the information is checked a few times and every division or part of the training dataset is get the opportunity to be utilized in the training as well as testing stages. When recorded our first result we applied the k-cross validation in order to be sure strength of train model. In cross validation strategy, the dataset was divided into 9 divisions. One is utilized for testing and 8 for training in the primary run. In the subsequent run, an alternate part is utilized for testing and 8 parts for training. The runs proceed until each part or division is allowed to be part of the training dataset and the testing data.

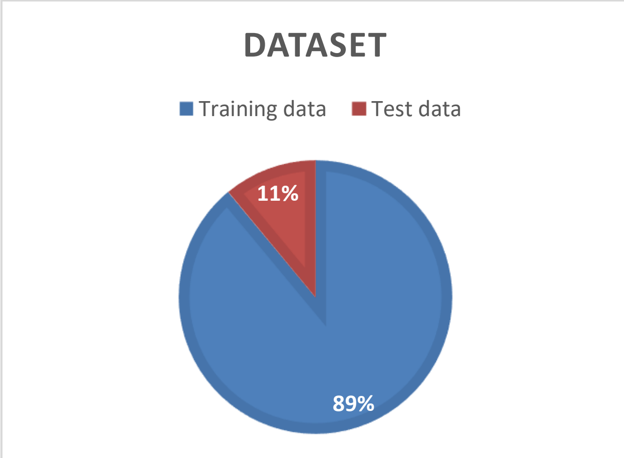

Figure 7. Cross Validation Division

2.7.1 Report Dataset after applied cross validation tests:

Unigram results

Table 3 Prediction result for test 1

	Precision	recall	f1-score	support
1	0.80	0.81	0.80	43
2	0.71	0.63	0.61	43
3	0.61	0.68	0.64	34
micro avg	0.71	0.71	0.71	120
macro avg	0.71	0.71	0.71	120
weighted avg	0.71	0.71	0.71	120

Table 4 prediction result for test 2

	Precision	recall	f1-score	support
1	0.33	0.50	0.40	6
2	0.97	0.89	0.92	96
3	0.54	0.68	0.60	19
micro avg	0.83	0.83	0.83	121
macro avg	0.61	0.69	0.64	121
weighted avg	0.87	0.83	0.85	121

Table 5 Prediction result for test 3

	Precision	recall	f1-score	support
1	0.41	0.64	0.50	14
2	0.87	0.77	0.82	81
3	0.39	0.42	0.41	26
micro avg	0.68	0.68	0.68	121
macro avg	0.56	0.61	0.57	121
weighted avg	0.72	0.68	0.69	121

Table 6 Prediction result for test 4

	Precision	recall	f1-score	support
1	0.78	0.76	0.77	37
2	0.79	0.74	0.76	61
3	0.48	0.59	0.53	22
micro avg	0.72	0.72	0.72	120
macro avg	0.68	0.70	0.69	120
weighted avg	0.73	0.72	0.72	120

Table 7 Prediction result for test 5

	Precision	recall	f1-score	support
1	0.72	0.83	0.77	65
2	0.50	0.32	0.39	19
3	0.53	0.49	0.51	37
micro avg	0.64	0.64	0.64	121
macro avg	0.58	0.54	0.56	121
weighted avg	0.63	0.64	0.63	121

Table 8 Prediction result for test 6

	Precision	recall	f1-score	support
1	0.77	0.81	0.79	67
2	0.82	0.61	0.70	23
3	0.53	0.58	0.55	31
micro avg	0.71	0.71	0.71	121
macro avg	0.71	0.67	0.68	121
weighted avg	0.72	0.71	0.71	121

Table 9 Prediction result for test 7

	Precision	recall	f1-score	support
1	0.77	0.81	0.79	67
2	0.82	0.61	0.70	23
3	0.53	0.58	0.55	31
micro avg	0.71	0.71	0.71	121
macro avg	0.71	0.67	0.68	121
weighted avg	0.72	0.71	0.71	121

Table 10 Prediction result for test 8

	Precision	recall	f1-score	support
1	0.60	0.75	0.67	24
2	0.81	0.66	0.72	32
3	0.85	0.85	0.85	65
micro avg	0.78	0.78	0.78	121
macro avg	0.75	0.75	0.75	121
weighted avg	0.79	0.78	0.78	121

Table 11 Prediction result for test 9

	Precision	recall	f1-score	support
1	0.67	0.76	0.72	38
2	0.92	0.69	0.79	49
3	0.56	0.68	0.61	34
micro avg	0.71	0.71	0.71	121
macro avg	0.72	0.71	0.71	121
weighted avg	0.74	0.71	0.72	121

Bigram results

Table 12 Prediction result for test 1

	Precision	recall	f1-score	support
1	0.78	0.65	0.71	43
2	0.60	0.86	0.70	43
3	0.55	0.35	0.43	34
micro avg	0.64	0.64	0.64	120
macro avg	0.64	0.62	0..61	120
weighted avg	0.65	0.64	0.63	120

Table 13 Prediction result for test 2

	Precision	recall	f1-score	support
1	0.23	0.50	0.32	6
2	0.88	0.79	0.84	96
3	0.32	0.37	0.37	19
micro avg	0.71	0.71	0.71	121
macro avg	0.48	0.55	0.50	121
weighted avg	0.76	0.71	0.73	121

Table 14 Prediction result for test

	Precision	recall	f1-score	support
1	0.46	0.86	0.60	14
2	0.87	0.88	0.87	81
3	0.69	0.35	0.46	26
micro avg	0.76	0.76	0.76	121
macro avg	0.67	0.69	0.64	121
weighted avg	0.78	0.76	0.75	121

Table 15 Prediction result for test 4

	Precision	recall	f1-score	support
1	0.84	0.70	0.76	37
2	0.72	0.79	0.75	61
3	0.27	0.27	0.27	22
micro avg	0.67	0.67	0.67	120
macro avg	0.61	0.59	0.60	120
weighted avg	0.67	0.67	0.67	120

Table 16 Prediction result for test 5

	Precision	recall	f1-score	support
1	0.79	0.85	0.81	65
2	0.48	0.68	0.57	19
3	0.62	0.41	0.49	37
micro avg	0.69	0.69	0.69	121
macro avg	0.63	0.65	0.62	121
weighted avg	0.69	0.69	0.68	121

Table 17 Prediction result for test 6

	Precision	recall	f1-score	support
1	0.78	0.78	0.78	67
2	0.51	0.83	0.63	23
3	0.47	0.26	0.33	31
micro avg	0.65	0.65	0.65	121
macro avg	0.59	0.62	0.58	121
weighted avg	0.65	0.65	0.64	121

Table 18 Prediction result for test 7

	Precision	recall	f1-score	support
1	0.47	0.83	0.60	30
2	0.30	0.78	0.64	27
3	0.83	0.38	0.52	64
micro avg	0.58	0.58	0.58	121
macro avg	0.61	0.66	0.58	121
weighted avg	0.67	0.58	0.56	121

Table 19 Prediction result for test 8

	Precision	recall	f1-score	support
1	0.44	0.71	0.54	24
2	0.57	0.72	0.64	32
3	0.69	0.45	0.54	65
micro avg	0.57	0.57	0.57	121
macro avg	0.57	0.62	0.57	121
weighted avg	0.61	0.57	0.57	121

Table 20 Prediction result for test 9

	Precision	recall	f1-score	support

1	0.57	0.68	0.62	38
2	0.65	0.76	0.70	49
3	0.56	0.29	0.38	34
micro avg	0.60	0.60	0.60	121
macro avg	0.59	0.58	0.257	121
weighted avg	0.60	0.60	0.59	121

Method explains research timeline, research design, research procedure (in the form of algorithms, Pseudocode or other), and data acquisition. The description of the course of research should be supported by references, so the explanation can be accepted scientifically.

3 RESULT AND DISCUSSION

Tweets were extracted from Twitter using web scrapper with help of Twitter advance search It seems that negative tweets was little bit higher compare to positive and neutral Tweets. Analyses was done on this marked datasets utilizing the term frequency–Inverse document frequency (TF–IDF) extraction procedure. We use the framework where the pre-processor is applied to the raw sentences which make it increasingly fitting to comprehend. The dataset collected was label to their respective polarities, positive, negative and neutral as shown in Figure 8.

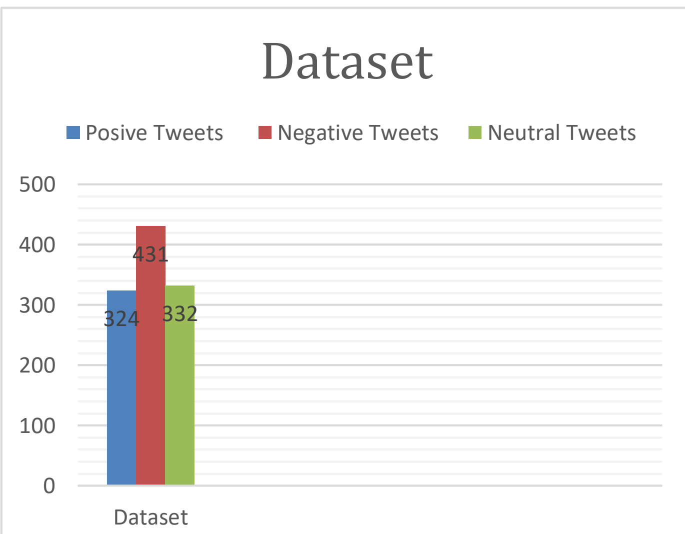

Figure 8. Emotion Distribution of Data Set.

Some tweets that we have collected was short, it was not possible to apply higher n-gram, since when n-gram length increases, and the number of time you will perceive any given n-gram will diminish. The drop of accuracy in bigram may cause sparsity. The more sparse data is, the more terrible you can train it. Thus, regardless of that a higher-request n-gram model, the more data in our context will contain and more will lead to over-fitting, a situation where your training data will memorizes instead of learning which will cause poor prediction, to avoid these situation we prefer to use only lower n-gram model.

We applied cross validation we divided almost 9 equal subsets as shown in Table 21 below, in order to reduce bias. We train the dataset on a subset and utilize the other subset to assess the model's performance. To decrease fluctuation we achieve various rounds of cross-validation with various subsets from the same dataset.

Table 21 Cross validation subsets

Train Data	967	966	966	967	966	966	966	966	966
Test Dara	120	121	121	120	121	121	121	121	121

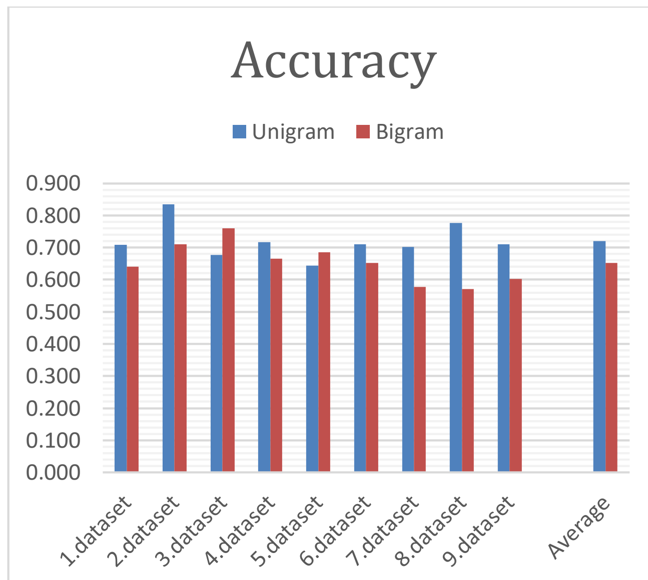

Figure 9. Emotion Distribution of Test Data Set Accuracy

1. Dataset is the test data, untrained dataset we obtained 70.8% of accuracy when used unigram compared when we used bigram 64.1% accuracy, 2 to 9 dataset is train dataset.

Our vocabulary was rich but most of our vocabulary wasn't have enough frequency with this reason our unigram perform better that our bigram model as shown figure 9 above.in most of time bigram perform better than unigram but in our case we were working with limit dataset since demonetization took place in short period we couldn't maintained to collect a huge data.

The overview of accuracy, recall and Precision of our dataset is as shown below. Table 22 when used unigram we obtained 71% and table 23 when we used bigram we got 65% accuracy.

Table 22 Unigram train data

	Precision	recall	f1-score	support
1	0.69	0.77	0.73	324
2	0.83	0.73	0.77	431
3	0.62	0.65	0.64	332
micro avg	0.72	0.72	0.72	1087
macro avg	0.71	0.72	0.71	1087
weighted avg	0.73	0.72	0.72	1087

Table 23 Bigram train data

	Precision	recall	f1-score	support
1	0.64	0.75	0.69	324
2	0.69	0.80	0.74	431
3	0.57	0.36	0.44	332
micro avg	0.65	0.65	0.65	1087
macro avg	0.64	0.64	0.63	1087
weighted avg	0.64	0.65	0.64	1087

The general output shows the unigram outperform better than when used bigram, for all result precision and f1-score. The prediction of neutral, except the recall of negative it gave us better result when applied bigram. Neutral prediction was perfect when we used a unigram compare when we used bigram where most of the time it assumed either polarities positive and negative.

This work originally was an extended version of our previous work on [26]. Here we use unigram and bigram as a classification feature. However, as shown before, accuracy was not improved significantly. In the previous work, we got 70.4% accuracy and the current work with the unigram feature we got 71% accuracy. As for the use of the bigram feature the accuracy was lower.

4 CONCLUSION

In this paper we perform Twitter sentiment analysis to understand people's opinions on demonetization. The gathered dataset which was extracted from Twitter using web scrapper. The data size was limited we had to work on small size and to avoid bias prediction we have applied cross validation. Multinomial Naive Bayes (MNB) algorithms are implemented as well as unigram and bigram as our feature. Analyses was done on this marked datasets utilizing the term frequency–Inverse document frequency (TF–IDF) extraction procedure. We use the framework where the pre-processor is applied to the raw sentences which make it increasingly fitting to comprehend and after test data was executed, we compare both unigram and bigram. We found our unigram perform better than our bigram model.

REFERENCES

[1] B. Liu, "Sentiment Analysis and Opinion Mining," Morgan & Claypool Publishers, 2012.

[2] "Demonetisation Decoded: A Critique of India's Currency Experiment - Jayati Ghosh, C. P. Chandrasekhar, Prabhat Patnaik - Google Books." https://books.google.co.id/books/about/Demonetisation_Decoded.html?id=FkYlDwAAQBAJ&redir_esc=y (accessed Sep. 03, 2020).

[3] "Twitter - Company." https://about.twitter.com/en_us/company.html (accessed Sep. 03, 2020).

[4] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment Analysis of Twitter Data," Association for Computational Linguistics, 2011. Accessed: Sep. 03, 2020. [Online]. Available: http://www.webconfs.com/stop-words.php.

[5] B. O'connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith, "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series." Accessed: Sep. 03, 2020. [Online]. Available: http://www.sca.isr.umich.

[6] A. Hernandez-Suarez, G. Sanchez-Perez, K. Toscano-Medina, V. Martinez-Hernandez, V. Sanchez, and H. Perez-Meana, "A Web Scraping Methodology for Bypassing Twitter API Restrictions | Request PDF." https://www.researchgate.net/publication/324055334_A_Web_Scraping_Methodology_for_Bypassing_Twitter_API_Restrictions (accessed Sep. 03, 2020).

[7] N. Godbole, M. Srinivasaiah, and S. Skiena, "Large-Scale Sentiment Analysis for News and Blogs." Accessed: Sep. 03, 2020. [Online]. Available: http://www.textmap.com/sentiment.

[8] M. Korayem, D. Crandall, and M. Abdul-Mageed, "Subjectivity and Sentiment Analysis of Arabic: A Survey," in *Communications in Computer and Information Science*, 2012, vol. 322, pp. 128–139, doi: 10.1007/978-3-642-35326-0_14.

[9] R. Mcdonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar, "Structured Models for Fine-to-Coarse Sentiment Analysis."

[10] J. Benhardus, "Streaming Trend Detection in Twitter," 2010. Accessed: Sep. 03, 2020. [Online]. Available: http://twitter.com.

[11] C. Strapparava and R. Mihalcea, "SemEval-2007 Task 14: Affective Text," 2007. Accessed: Sep. 03, 2020. [Online]. Available: https://www.aclweb.org/anthology/S07-1013.

[12] B. Das Sarit Chakraborty Student Member and I. Member, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation."

[13] E. Demirtas and M. Pechenizkiy, *Cross-lingual Polarity Detection with Machine Translation*. 2013.

[14] E. Frank and R. R. Bouckaert, "Naive Bayes for Text Classification with Unbalanced Classes."

[15] A. Mccallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification."

[16] M. Abbas, K. Ali, S. Memon, A. Jamali, S. Memon, and A. Ahmed, "Multinomial Naive Bayes Classification Model for Sentiment Analysis," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 19, no. 3, pp. 62–67, 2019, Accessed: Sep. 03, 2020. [Online]. Available: https://www.researchgate.net/publication/334451164_Multinomial_Naive_Bayes_Classification_Model_for_Sentiment_Analysis.

[17] A. R. Susanti, T. Djatna, and W. A. Kusuma, "Twitter's sentiment analysis on GSM services using Multinomial Naïve Bayes," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 15, no. 3, pp. 1354–1361, 2017, doi: 10.12928/TELKOMNIKA.v15i3.4284.

[18] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in *Lecture Notes in Artificial*

Intelligence (Subseries of Lecture Notes in Computer Science), 2004, vol. 3339, pp. 488–499, doi: 10.1007/978-3-540-30549-1_43.

[19] P. Chaovalit and L. Thou, "Movie review mining: A comparison between supervised and unsupervised classification approaches," in *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2005, p. 112, doi: 10.1109/hicss.2005.445.

[20] P. K. Srijith, S. Shevade, and S. Sundararajan, "Semi-supervised Gaussian Process Ordinal Regression."

[21] G. I. Webb, M. J. Pazzani, and D. Billsus, "Machine Learning for User Modeling."

[22] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision." Accessed: Sep. 03, 2020. [Online]. Available: http://tinyurl.com/cvvg9a.

[23] Y. HaCohen-Kerner and H. Badash, "Positive and Negative Sentiment Words in a Blog Corpus Written in Hebrew," in *Procedia Computer Science*, Jan. 2016, vol. 96, pp. 733–743, doi: 10.1016/j.procs.2016.08.257.

[24] A. C. E. S. Lima, L. N. De Castro, and J. M. Corchado, "A polarity analysis framework for Twitter messages," *Appl. Math. Comput.*, vol. 270, pp. 756–767, Nov. 2015, doi: 10.1016/j.amc.2015.08.059.

[25] B. Hegde, N. H. S, and M. Prakash, "Sentiment analysis of Twitter data: A machine learning approach to analyse demonetization tweets," *Int. Res. J. Eng. Technol.*, 2018, Accessed: Sep. 03, 2020. [Online]. Available: www.irjet.net.

[26] I. M. Noor, "Sentiment Analysis on New Currency in Kenya Using Twitter Dataset," in *Proceeding International Conference on Science and Engineering*, 2020, pp. 237–240, Accessed: Sep. 04, 2020. [Online]. Available: http://sunankalijaga.org/prosiding/index.php/icse/article/view/503/478.