International Journal of Research Studies in Science, Engineering and Technology
Volume 7, Issue 9, 2020, PP 07-15
ISSN 2349-476X

Sentiment Analysis of Afaan Oromo using Machine learning Approach

Megersa Oljira

Ambo University, Ethiopia

***Corresponding Author:** *Megersa Oljira, Ambo University, Ethiopia*

ABSTRACT

The evolution of social media and social networks provide people with unprecedented opportunities to express and share their thoughts, views, opinions and feelings about almost anything through their personal webpages and blogs or using social network sites like Facebook, Twitter, and Blogger. This study focuses on sentiment analysis of Afan Oromo social media content because automatically identifying and classifying opinions from social media posts can provide significant economic values and social benefits. We employed Naïve Bayes machine learning algorithm and different n-grams such as unigram, bigram, trigram and their combinations as features. The experiment results show that, among the six different learning setups, the accuracy of the unigram+bigram approach is found to be promising.

Keywords: *Afan Oromo, Sentiment Analysis, Machine Learning*

INTRODUCTION

Sentiment analysis is deals with the classification of people's opinion in to correct classes. In the present-day scenario, social media play a pertinent role in providing information about any product from different reviews, blogs, and comments. In order to derive meaningful information from people's sentiments, different machine learning techniques are applied by scholars and practitioners [1].

Sentiment analysis can be conducted at different levels. The most famous level of sentiment analysis is document level, aspect level, and sentence level. The document level deals with determining the overall opinion of the document expressed by the opinion holder [1]. The aspect level sentiment analysis is the task of extracting the relevant aspects of the reviewed product or entity and determining the sentiment of the corresponding opinion about them. Whereas, Sentence level sentiment analysis is the more fine-grained analysis of the document. In this, polarity is calculated for each sentence as each sentence is considered a separate unit and each sentence can have different opinions.

In this study, we considered the document-level sentiment analysis assumes that a document expresses a single sentiment. This approach suitable for some areas like reviews, where a last statement about the entity is assumed to be required which is a weighted conclusion arising from different sides even if the review carries different opinions. The social networking site Facebook is the targeted website for this paper. This is because Facebook has many members and vast user-generated data is available. Thus, this research focuses on sentiment analysis of Afan Oromo texts on facebook. In this study, we developed Naïve Bayes machine learning approach. The online text is mostly in the text format and unstructured in nature. Thus, the stop words and other unwanted characters and information are removed from the comments for further analysis. These comments go through a process of vectorization in which, the text data are converted into matrix of numbers.

These matrices are then given input to machine learning techniques for classification of the comments. Different parameters are then used to evaluate the performance of the machine learning algorithm. The main contribution of the paper can be stated as follows:

1. Naïve Bayes machine learning algorithm is proposed for the classification of Afan Oromo sentiment using n-gram techniques viz., Unigram, Bigram, Trigram, combination of unigram and bigram, bigram and trigram, and unigram and bigram and unigram and trigram.
2. The performance of the proposed technique is evaluated using parameters like precision, recall, f-measure, and accuracy.

RELATED WORK

In Afaan Oromoo the sentiment analysis is new and only a few works were studied. We encountered only two researchers on the Afaan Oromoo language. [2], conducted aspect-based summarization of Afaan Oromoo news text on the news domain. This work is the first attempt at Afaan Oromoo opinion mining. The researchers used manually crafted rules and a lexicon-based approach. The dataset obtained from the ORTO news service. As reported by the researcher, even though the system shows good results, the lack of resources such as lexical database and linguistic resources such as POS made the work challenging. There are also gaps that are needed to be elaborated more. For example, people express their feeling on social media indirectly and their system cannot handle this problem. The other works by [3]. The researcher developed an unsupervised approach for Afaan Oromoo on a Facebook domain. Data is obtained from the official facebook page of the Oromoo democratic page and other Activists pages on current political situations. N-gram and POS used as features. As the researcher claims the proposed work shows a promising result.

Table1. *Summary of SA and opinion mining of AO*

Author	objective	features	Model	classes	Selected model	Domain Afaan oromoo Data source	No Dataset
[2], Msc Thesis, DBU	Assign feature sentiment & summarization	Lexicons +rule based	rule based	+ve&-ve	General lexicon with rule based	ORTO news service	400 reviews
[3] Journal Article	Feature extraction & polarity classification	Lexicon, POS, N-gram	Unsupervised	+ve,-ve & neutral	bigram	(OPDO) official Facebook page, political bloggers page	600 reviews

The general work proposed by the two researchers needs the lexical database and it involves the manual collection of lexicons. Moreover, the machine learning method performs better with less human intervention [4]. In addition, regarding social media texts where nature the texts are informal, indirect [2] , slang and idiomatic it is difficult to deal with the previous techniques. According to the literature [5], the lexicon-based models were not very accurate and a good rule-based model was very hard to elaborate, we implemented state-of-the-art methods for Afan Oromoo sentiment analysis.

The naïve Bayes classifier is operates based on the Bayes theorem. The simplicity and efficiency, as well as the performance of the NB classifier, attracted many researchers to use it for sentiment analysis and text classification domains. The authors [6], [7], [8], [9] reported that NB is the most prominent algorithm for sentiment analysis and text classification tasks despite its simplicity. It is a probabilistic approach integrating the Bayes' algorithm [10] that allows computing the probability of features belonging to a label:

[6] Have studied the aspect of sentiment classification based on categorization study, with binary classification (positive and negative) sentiments. The authors made an experiment with three different machine learning algorithms, such as NB, SVM, and ME. The classification process is performed using the n-gram technique like unigram, bigram, and combination of both unigram and bigram. The researchers also have used bag-of- word features framework to implement the machine learning algorithms. [11] Implemented naïve Bayes and maximum entropy for twitter data. The authors used TF-IDF and Count Vectorizer feature extraction methods. Based on their experiment the naïve Bayes performs better results with TF-IDF and n-gram

THE PROPOSED WORK

The proposed sentiment analysis model is depicted in Figure 1. This model has a number of components for preprocessing, Stop word filtration, stemming, n-gram tokenization, tf-idf weighting and splitting into training and testing. The upcoming sections describe the various components of the model.

Data Collection

For this study, the primary data source from *Oromo Democratic Party /ODP* official Facebook page is extracted by using face graph

API. The reason for choosing this page is that there is a huge user generated opinion. This page is the government organization page and the government policy related post is released every day on this page.

So that the genuine and reliable user-generated data is available on this page. Moreover, this page is a public page and people express their idea about government freely on this page. We focused on sociopolitical related issues, government policy, and other related issues.

The total amount of reviews collected is 1452, 726 positive and 726 negatives. The extracted data is saved in comma delimiter (CSV) format in excel.

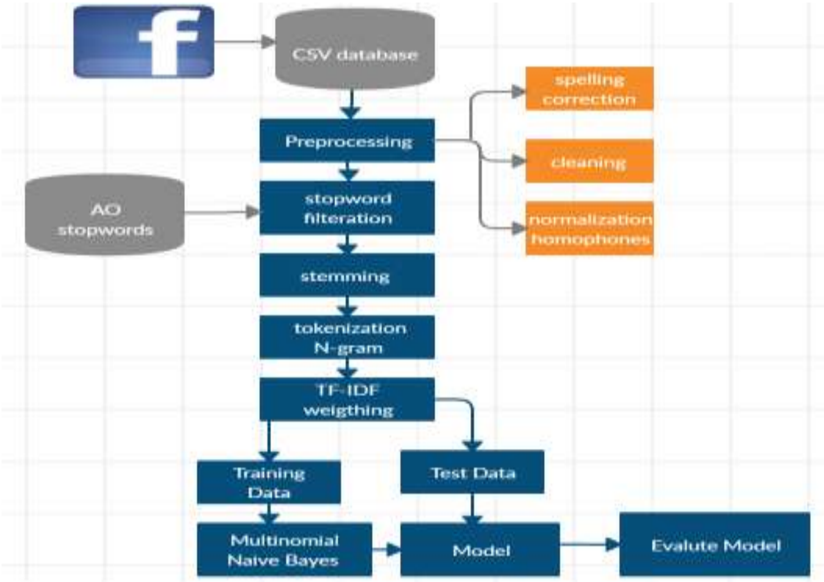

Figure1. Architecture of the Proposed work

Data Preprocessing

As stated previously, for this thesis we used a supervised machine learning method. Since the supervised method requires the labeled dataset for training purposes the dataset collected was labeled manually by experts. After that, the data is split into training and testing data using scikit-learn train_test_split. The training data used to train the classifiers, and the test data used for testing the accuracy of the classifiers.

We split our dataset according to the 80/20 rule [7] i.e. eighty percent of the dataset goes to the training set and twenty present goes to the test set. We used the train_test_split method of the sklearn library to perform this task in python. train_test_split is faster, simpler so it would be easier to analyze the testing errors.

Another step is preprocessing in order to exclude irrelevant data from the dataset. Preprocessing is very import as it reduces the computational time and increases the classifier performance because noisy data can slow the learning process and decrease the efficiency of the system. Accordingly, our preprocessing includes the following:

Cleaning: Removal of user names**,** Removal of links**,** lower casing**,** Removal of none Afaan Oromoo texts, unnecessary characters, etc.

Stopword Removal: some Afaan Oromoo Stopwords are significant for the sentiment classification and need to remain in the text. For instance, the “hin” is used to indicate the negativity of the word: for example, “dhufeera”, “hin dhufne”. In another case, some stop words constitute a phrase: “walii hin gallu”, “isin waliin jirra” etc. These stop words portray important information. So we filtered removed stop words through a manual process that is not relevant for the classification process.

Normalization: Homophones like “baay’ee” and “baayyee” has the same meaning with different writing. The only difference is that the apostrophe “’” is replaced by “y”.

- Normalization of elongated texts, for example, sin jaallannaaaaaa is normalized to sin jaallanna
- Normalization of numbers into equivalent texts. Example:”sin jaallanna 100% “normalized to” sin jaallanna persentii dhibba tokko”.

Spelling Correction: we encounter many wrongly spelled texts. So they need to be corrected to the right spelling.

Tokenization

Tokenization is the process of splitting an input sequence into tokens. It a useful unit for semantic processing. The tokens usually consist of either a single word or what is called an N-gram, meaning that N consecutive words are split into a single token. The idea is to preserve some of the information that is stored in the order of the words.

Stemming

Stemming is the technique of reducing inflection in words. For the purpose of this study, we used the stemmer developed by [12]. Stemmer is the system that reduces morphological variants of words into base or root form. In morphologically rich languages like Afaan Oromo, a stemmer will lead to important improvements in sentiment Analysis [2].

Feature Selection

Features are the portion of the information we take from the text and feed to the algorithm for classification. In sentiment analysis, feature selection is the process of selecting a specific subset of the terms of the training set and using only them in the classification algorithm. The task of the feature selection process is performed before the training of the classifier. For this thesis, the TF-IDF feature selection is employed for the Naïve Bayes model [13], [14], and [15]. Term frequency and inverse document frequency (TF-IDF), is a simple and effective feature selection approach. It removes from the original feature space the infrequent terms that are considered as less or non-informative for classification.

The importance or informativeness of the feature of a term is calculated based on both term frequency and inverse document frequency.

In addition to that, we used n-gram as a feature. Using only single words as features have several disadvantages. N-grams are better for negation handling and handle the context of the word.

The TF-IDF feature selection is a simple and effective feature selection paradigm. It is easy to remove hundreds of rare and uninformative features. In TF-IDF weighting the importance of features is computed based on both term frequency and inverse document frequency.

TF-IDF Which Stands for Term Frequency – *Inverse Document Frequency* and it is a statistical method of evaluating the significance of a word in given documents.

Term frequency (tf): measures the frequency of the occurrences of a given term in a document.

Inverse document frequency: measures the cost of the word in the document, i.e. if the word is common or infrequent in the entire document. The main idea behind tf-idf follows that the terms that appear frequently in a document are less important than terms that are infrequent or rare. The TF-ID customs the vector space modeling system for text document representation.

TF-IDF measure accounts both, the Term Frequency and the Inverse Document Frequency. The term frequency indicated as $tf(t,d)$ is the total number of times a given term *t* appears in document *d* against the total number of all words in the document. If the frequency of a given term increases in a document, then its *tf* also increases. The inverse document frequency represented as $idf(t,D)$ is a measure of how much information the word provides. In another word, it measures the worth weight of a given word in a given document. IDF shows how common or rare a given word is across all documents.

Term frequency–Inverse document frequency is the product of term frequency $tf\,(t,d)$ and inverse document frequency$idf(t,D)$. TF-IDF can be computed as:

$$TF(t,d) = \frac{number\ of\ times\ term\ (t) appears\ in\ document\ (d)}{total\ number\ of\ terms\ \ (t)\ in\ document\ (d)}$$

The Inverse Document Frequency measures how word contribute to discriminate each document and obtained as the following,

$$IDF(t,D) = log\left(\frac{total\ number\ of\ Documents\ (D)}{number\ of\ documents\ with\ term\ (t)\ in\ it}\right)$$

Now we got TF and IDF, and next, we can calculate TFIDF as below.

$$TFIDF(t,d,D) = TF(t,d).IDF(t,D)$$

The second feature we used is n-gram. N-grams are the combination of neighboring words or letters of length *n* that can be searched in the source text.

Features can take the form of unigrams, bigrams or n-grams subject to requirements or hybrid of them [6], [7]. In some cases the studies find that better polarity classification can be achieved by using bigrams and unigrams instead of trigrams, others find that bigrams outperform unigrams in sentiment classification [7].

N-gram is an effective feature selection approach. For example when using 2-grams, the semantic difference between "cat calling" and "cat food" would not get lost, which it would be when using 1-grams [11].

In most literature, single words were used as features. If the term “gadhee or yaraa” ("bad”) occurs in a document, it is likely to have a negative sentiment. However, considering only single words as a feature have several limitations.

For example, Negations, such as “gadhee miti” ("not bad") or (gaarii miti) "not good" for example, will not be taken into account. So this leads to misclassification as a single word features can even change the direction of the classification.

Negations are used in order to contradict or deny a statement. Commonly used Afaan Oromoo negations are introduced with ’hin’ and ’miti’. An example with a sentiment bearing verb: "inni ni danda'a “(he can) "inni hin danda’u" (he cannot) An example with a sentiment bearing adverb: "kun gaariidha" (this is good) "kun gaarii miti" (this is not good). So in this thesis, we used n-grams (1, 2, and 3) as features in addition to single words can overcome this problem.

In another case, Phrasal verbs are common in the Afaan Oromoo language. They are verbs which are composed of one verb in combination with one or more particles, for example, "waa'ee saa walii hin gallu" (we disagree about it). Each part needs to be taken into account for the expression to make sense. To overcome such problems, we used n-grams as features in addition to single words. [1]. The detail architecture of the proposed MNB is depicted in figure 1.

1

Application of Naïve Bayes Classifier

Naïve Bayes classifier is widely used in text classification domain and sentiment analysis due to its simplicity and efficiency [6], [14], [16], [17].

$$P(features) = \frac{P(label) * P(features|label)}{P(feaatures)}$$

Where $P(label|features)$ is the posterior probability of features belonging to a label (positive or negative), $P(label)$ is the prior probability of a given label, $P\ (features|label)$ is the conditional probability that the particular feature in features appears given label, $P\ (features)$ is the prior probability of the feature in features.

To illustrate the ‘naïve’ assumption that the features are independent of each other. That gives the following:

$$p(features) \approx p(label) * p(label)p(label) * \dots * p(label) = \prod_{i=1}^{n} p(fi|label)$$

$$p(features) = \frac{p(label) * \prod_{i=1}^{n} p(fi|label)}{p(feaatures)}$$

f_i =indicates an individual feature.

Although the Naïve Bayes model assumes that features are generated independently of their positions, it still gives good results in real tasks. The main goal of the classification is to define the label the feature belongs to. Therefore, we do not interest in finding the probability itself, however, the most probable label has to be defined. Naïve Bayes classifier uses the maximum a posteriori (MAP) estimation to define the most probable label $label_{map}$, [14], [10].

$$label_{map} = arg\max_{label\ \in L} \left[\frac{\hat{P}(label) * \prod_{i=1}^{n} \hat{P}(fi|label)}{\hat{P}(feaatures)}\right]$$

Since the Denominator is common it can be omitted and simplified as: Hence expression for computing $label_{map}$ can be rewritten as:

$$label_{map} = arg\max_{label\ \in L} \left[\hat{P}(label) * \prod_{i=1}^{n} \hat{P}(fi|label)\right]$$

P marked as $\hat{P}$ because the truth values of the corresponding parameters will be estimated from the training dataset.

In this thesis work Multinomial Naïve Bayes classifier is applied (features show how many times each word occurs in the given dataset). As discussed earlier the multinomial Naive Bayes classifier is well suit for classification with discrete features like text classification task [2]. Multinomial NB utilizes the naive Bayes algorithm for multinomially dispersed data and is one of the two typical naive Bayes variants used in text classification (where the data are classically represented as word vector counts and although tf-idf vectors are also known to work well in practice) [3]. The distribution is parameterized by vectors $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ for each class y, where n is the number of features this indicates in text classification, the size of the vocabulary and θyi is the probability $P(x_i \mid y)$ of feature i appearing in a sample belonging to class y. Our training method consists of relative-frequency estimation of P (label or class) and P (fi | class), using add-one smoothing which is called Laplace smoothing.

The parameters θy is calculated by a smoothed version of maximum likelihood, i.e. relative frequency counting:

$$\hat{\theta} yi = \frac{N_{yi} + \alpha}{N_y + \alpha n}$$

Where $N_{yi} = \sum_{x \in T} \quad xi$ is the number of times feature i appears in a sample of class y in the training set T, and $N_y = \sum_{i=1}^{n} \quad N_{yi}$ is the total count of all features for class y.

The smoothing priors $\alpha \geq 0$ stands for features not present in the learning samples or in our case vocabulary and prevents zero probabilities in further computations [4]. Accordingly, in this thesis, we implemented our experiment by using a sklearn feature extraction python library.

Results and Discussion

This chapter introduces the results obtained when conducting an experiment by using the classifier Naïve Bayes (MNB). The naive Bayes algorithm implemented using the sci-kit-learn library in python. We used evaluation metrics (precision, recall, accuracy and f1 score) to evaluate the performance of the classifier on proposed work.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \ldots\ldots (1)$$

TP is the number of true positives: the reviews/comments that are actually positive and estimated as positive.

TN is the number of true negatives: the reviews/comments that are actually negative and estimated as negative, *FP is the number of false positives*: the reviews/comments that are actually negative but estimated as positive,

FN is the number of false negatives: the reviews/comments that are actually positive but estimated as negative.

A Precision can be estimated using the following formula [18]:

$$precison = \frac{TP}{TP+FP} \ldots (2)$$

Precision shows how many positive reviews received from the classifier are correct. The greater precision the fewer number of false hits. However, precision does not show whether all the correct answers are returned by the classifier. In order to take into account, the latter recall will be used [18]:

$$recall = \frac{TP}{TP+FN} \ldots (3)$$

Recall shows the ability of the classifier to "guess" as many correct answers, (reviews with correct labels) as possible out of the expected.

The more precision and recall the better. However, simultaneous achievement of high precision and recall is almost impossible in real life that is why the balance between two metrics has to be found. $F1$ score is a harmonic mean of precision and recall [18]:

$$f_1 = \frac{2 * preecisoin \ * recall}{precision \ + recall} \ldots (4)$$

Implementation of Multinomial Naïve Bayes

After preprocessing of the comments or text reviews, we need to convert to a numeric vector matrix. We used TF-IDF Vectorizer to perform this. This is performed by using sckit-learn python class. TF-IDF is word frequency scores that try to suggests words that are more interesting, e.g. frequent in a document but not across documents. The Tfidf Vectorizer tokenize comments with different n-gram, learn the vocabulary and inverse document frequency weightings. After this step is completed different experiment is performed using unigram bigram, trigram, and their combination.

The confusion matrix and various evaluation parameters such as precision, recall, f-measure, and accuracy values obtained after classification

[2] *https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.Multinomial NB.html*
[3] *https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.Multinomial NB.html*
[4] https://scikit-learn.org/stable/modules/naive_bayes.html

using Multinomial Naïve Bayes n-gram techniques are shown in the classification report below. It can be analyzed that the accuracy value obtained using the hybrid of unigram+bigram, unigram+trigram is better than the value obtained using techniques such as bigram and trigram. NB method is a probabilistic method, where the features are independent of each other. Hence, when the analysis is carried out using unigram and the combination (unigram and bigram) or (unigram and trigram), the accuracy value obtained is comparatively better than that obtained using bigram and trigram. The accuracy obtained by using the trigram technique is the lowest accuracy. However, it is possible to extract the most informative phrases from the text. For example: ‘isin wajjiin jirra’, ‘baayyee namatti toltu”, ‘wow namatti tola’ these like phrases can be extracted by using trigram technique. Using only the trigram techniques decreases the accuracy because trigram is being considered for analysis of features, words are repeated a number of times; thus, it affects the probability of the document. For example: ‘hamaa keessan dhagahuu hin barbaadu’, when this phrase tokenized into trigram, ‘hamaa keessan dhagahuu’, ‘keessan dhagahuu hin’, ‘dhagahuu hin barbaadu’, ‘dhagahuu hin barbaadu’ shows negative polarity, whereas the text or sentence represents positive sentiment. That is why the accuracy of the trigram is decreased than the others. So, this experiment reveals that using the hybrid of unigram and bigram is the best technique for Afaan Oromoo sentiment analysis. When the hybrid of bigram and unigram or trigram and unigram is used, the unigram techniques extract individual word feature and the bigram or trigram technique extracts informative phrases in the text.

Implementation of Multinomial Naïve Bayes Classifier

Table2. *Confusion matrix, evaluation Metrics and accuracy for Naive Bayes n-gram classifier*

Method		Confusion Matrix		Evaluation Metrics			Accuracy
unigram		**Correct Labels**					
		Positive	Negative	Precision	Recall	f-measure	
	Positive	133	14	91	90	91	90.7
	Negative	13	131	90	91	91	
Bigram		70	8	48	90	62	71.1
		76	137	94	64	77	
	Positive	Positive	Negative	Precision	Recall	f-measure	
Trigram	Negative	14	0	10	100	17	54.6
		132	145	100	52	67	
		Positive	Negative	Precision	Recall	f-measure	
Unigram+Bigram	Positive	139	14	95	91	93	92.7
	Negative	7	131	90	95	93	
		Positive	Negative	Precision	Recall	f-measure	
Unigram+Trigram	Positive	139	15	95	90	93	92.4
	Negative	7	130	90	95	92	
		Positive	Negative	Precision	Recall	f-measure	
Bigram+Trigram	Positive	80	7	55	92	69	75
	Negative	66	138	95	68	79	

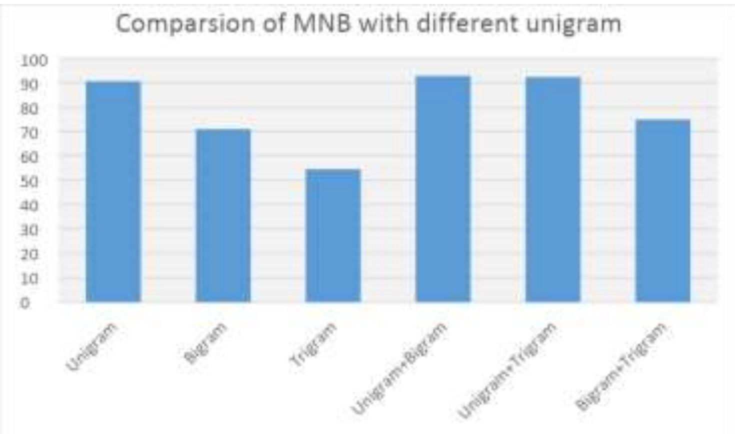

Figure2. *Comparison of the MNB with different n-gram techniques*

Conclusion and Future Work

The wide explosions of social media networks like Facebook, twitter, etc. provides a variety of benefits, in facilitating the way people share their opinion and increase the speed of public comments. Due to this, companies and governments receive high volumes of electronic comments every day. Identifying the polarity of the comments may be valuable input for decision making. Though, a large number of reviews make it difficult for a company or any institutions to react to the opinions rapidly and take appropriate decisions. Therefore, sentiment Analysis has become a major area of research interest in the field of Natural Language Processing and Text Mining to overcome these problems. The sentiment analysis task is under research since the early 2000s. Nevertheless, it is a new area and at an initial state in Afaan Oromoo. In this paper, we presented MNB machine learning approach to sentiment analysis on Afaan oromoo language. We used the for the machine learning task of sentiment analysis, we managed to collect around 3000 posts comments from different facebook pages. Our propose MNB approaches achieved According to the experiment, the result shows that accuracy of 90.7%, 71.1%, 54.6%, 92.7%, 92.4%, and 75% for unigram, bigram, trigram, unigram-bigram, unigram-trigram and bigram-trigram respectively.

- Online texts, especially social media networks like facebook may contain a lot of spelling mistakes, so spelling corrector can be applied to exclude errors.
- In this study considered only texts, Emoticons and emoji expressions that convey laugh, sad, angry, and happy, love, etc. need to be included and labeled whether emoticon, emoji expression refers to a positive or negative meaning.

References

[1] B. Liu, Sentiment analysis and opinion mining. Synthesis lectures on human language technologies, Morgan & Claypool Publishers, 2012.

[2] W. Tariku, "SENTIMENT MINING AND ASPECT BASED SUMMARIZATION OF OPINIONATED AFAAN OROMOO NEWS TEXT," DEBRE BERHAN UNIVERSITY, Debre Berham, 2017.

[3] J. Abate, "Unsupervised Opinion Mining Approach for Afaan Oromoo Sentiments, Haramaya University, 2019," 2019.

[4] G. Vinodhini and R. Chandrasekaran, "Sentiment analysis and Opinion Mining:A survey.," International Journal of Advanced Resarch in Computer Science and Software Engineerin, Vols. vol. 2, no. 6, pp. pp 1-11, 2012.

[5] Hailong, Wenyan and Bo, "Machine learning and lexicon based methods for sentiment classification: A survey.," *In Web Information System and Application Conference (WISA), 2014 11th,* vol. IEEE., pp. 262-265, 2014.

[6] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics.*, 2002.

[7] W. Philemon and W. Mulugeta, "A Machine Learning Approach to Multi-Scale Sentiment Analysis of Amharic Online Posts," *HiLCoE Journal of Computer Science and Technology, No. 2,* vol. Vol. 2, p. 87, 2014.

[8] Gautam and Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," in *Contemporary computing (IC3), 2014 seventh international conference,* 2014.

[9] Martin, D. Jurafsky and J. H., "Speech and Language Processing," in *Speech and Language Processing*, 2018, p. 21.

[10] Thakkar and Patel, "Approaches for sentiment analysis on twitter: A state-of-art study," in *arXiv preprint arXiv:1512.0104,* 2015.

[11] A. Tripathy, A. Agrawal and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Systems With Applications*, pp. 117-126, 2016.

[12] D. Tesfaye and E. Abebe, "Designing a Rule Based Stemmer for Afaan Oromo Text," *International Journal of Computational Linguistics (IJCL),* vol. 1, no. 2, pp. 1-11.

[13] Mossie et al., "Social network hate speech detection for Amharic language," *international journal of Computer Science and IT,* p. pp. 41–55, 2018.

[14] W. Medhat, A. Hassan and H. Korashy, "Sentiment analysis algorithms and applications: A survey.," *Ain Shams Engineering Journal,* vol. 5, pp. 093-1113, 2014.

[15] A. Bakliwal, J. Foster, J. Foster, J. v. d. Puil, R. O'Brien, L. Tounsi and M. Hughes, "Sentiment Analysis of Political Tweets," in *Towards an Accurate Classifier, Proceedings of the Workshop on Language in Social Media (LASM 2013, Atlanta, Georgia.*, Atlanta Georgia, 2013.

[16] C. C. Aggarwal and C. Zhai, *A survey of text classification algorithms,* USA: Springer US, 2012.

[17] B. Tang, S. Member, S. Kay, Fellow, H. He and S. Member, "Toward Optimal Feature Selection

in Naive Bayes for Text Categorization," IEEE, 2016.

[18] Jumayl, Robert and Lu, "Multi-Channel Convolutional Neural Network for Twitter Emotion and Sentiment Recognition," in *Proceedings of NAACL-HLT*, 2019.

[19] Bibi and Bahria, "Sentiment Analysis at Document Level, University Islamabad, Pakistan," ResearchGate.

[20] W. Medhat, A. Hassan and H. Korashy, "Sentiment analysis algorithms and applications: A survey.," *Ain Shams Engineering Journal*, vol. 5, pp. 093-1113, 2014.

[21] A. Bakliwal, J. Foster, J. v. Puil, R. O'Brien, L. Tounsi and M. Hughes, "Sentiment Analysis of Political Tweets," in *Towards an Accurate Classifier, Proceedings of the Workshop on Language in Social Media (LASM 2013, Atlanta, Georgia.*, 2013.

[22] Wang, Mossie and Jenq-Haur, "Social network hate speech detection for Amharic language," *international journal of Computer Science and IT,* p. pp. 41–55, 2018.

[23] Popescu and Etzioni, "Extracting product features and opinions from reviews," in *In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processin*, 2005.

Citation: *Megersa Oljira, "Sentiment Analysis of Afaan Oromo using Machine learning Approach" International Journal of Research Studies in Science, Engineering and Technology, 7(9), 2020, pp. 07-15.*