A Ternary Sentiment Classification of Bangla Text Data using Support Vector Machine and Random Forest Classifier

Partha Chakraborty, Farah Nawar, and Humayra Afrin Chowdhury

Abstract Sentiment analysis refers to the extraction of the underlying sentiment or emotion associated with an opinion. It is a part of the difficult field in the processing of natural languages (NLP) with several applications in the business, education, and political sectors. Unlike other languages, the amount of research performed on opinion mining for Bangla text is very small. There is also a lack of proper NLP tools for Bangla text processing. To bridge this gap, in this manuscript, a classification system has been developed for predicting the polarity of Bangla text data (i.e., positive, neutral, negative) using the two most efficient algorithms SVM and random forest for opinion mining. In this study, we experimented with unigrams, bigrams, and trigrams to illustrate how contextual information affects the overall performance of the classifiers. The dataset we used in this paper is imbalanced, which resembles natural characteristics of opinions in day-to-day life.

Keywords Opinion mining · Classification · Sentiment · *N*-gram · Machine learning

1 Introduction

The tremendous amount of unstructured data that is available online can be used for sentiment analysis. It helps to analyze customer's feedback to improve their products and services or to make decision in political movement or public welfare. There have been very few research carried out in sentiment analysis of Bangla text data in comparison with other languages (e.g., English, Spanish, French). In our paper, we have suggested an Bangla sentiment classification system which works effectively

In our paper, we used both tokenization and stemming of words in combination with *n*-gram (i.e., unigram, bigram, and trigram) for feature selection after removing

P. Chakraborty (✉) · F. Nawar · H. A. Chowdhury
Department of Computer Science and Engineering,Comilla University, Cumilla 3506, Bangladesh
e-mail: partha.chak@cou.ac.bd

J. K. Mandal et al. (eds.), *Topical Drifts in Intelligent Computing*, Lecture Notes in Networks and Systems 426, https://doi.org/10.1007/978-981-19-0745-6_8

the noise and redundancy of the initial raw data through several steps of preprocessing. Tokenization has shown better results than stemming with 68% $F1$-score. N-grams determine the relationship between $N-1$ neighboring words by assigning probabilities to them. It captures contextual information based on the value of n. The higher value of N ensures more contextual information from the sentences. N-gram techniques were used along with the TF-IDF method for vectorization. The processed data was fed into a classification model for evaluation. Our goal is to provide a system of classification that will be able to classify Bangla text into a category of positive, negative, and neutral classes using the two most efficient machine learning algorithms (i.e., SVM and RF) in opinion mining along with n-gram techniques and show a comparative analysis between the classifiers and n-grams.

The remaining sections of the manuscript are divided into four section. Section 2 consists of related works. Section 3 states the overall architecture and methodology of the system. The evaluation of the system and comparisons among the features are showed in sect. 4. Finally, Sect. 5 summarizes the findings and points the work's future directions.

2 Related Work

Over more than 1300 years, the Bengali language has been affected by a diversity of languages and cultures. There are also a few NLP tools for Bengali. Many experts have studied the Bengali language to fill this deficit. Abinash et al. [1] provided a resemblance of results obtained using the classification algorithms of NB and SVM. The algorithms help to decide whether a sentimental assessment is positive or negative. Taher et al. [2] focused their views on Bangla texts by using diverse web-based data. The ML process and the N-gram approach used to categorize Bangla papers were 89% precision achieved using 1–2 grams with SVM. Several works have been done on this work [3, 4]. Lee et al. [5] used the classification of Naive Bayes, SVM and maximum entropy. Pundlik et al. [6] suggested a strategy for categorizing Hindi speech documents into many classes using ontology, combining HSWN and LM classifier. To identify the best combination of user input, Rahman et al. [7] examined the effects of extractive function approaches. Uni, bi, and trigram are used for representations with TF-IDF independently. Tripto et al. [8] created deep learning-based models. They test the model's performance using a fresh corpus of English, Roman, and Bangla feedback from YouTube, and their method achieved 65.97% and 54.24% accuracy in three and five label sentiments, respectively. Related several works have been done in this work [9, 10]. It was developed by Haque et al. [11] to classify feedback using machine learning algorithms where SVM's accuracy is measured at 75.58%., according to a comparison of vectorizers.

3 Methodology

This section provides an overview of the overall system. Different preprocessing procedures reduced noise and redundancy from the underlying raw data. TF-IDF was utilized for vectorization and tokenization. In the training set, SVM and RF classifiers were used to classify the data. A three-category categorization strategy was used to organize the data (positive, neutral, and negative). The model was evaluated based on test results. Our model's system structure is depicted in Fig. 1.

3.1 Data Description

In this research, we used the dataset on Bangla news comments by Ashik et al. [12]. The dataset contains a total of 13802 Bangla text labeled with five categories: positive, slightly positive, neutral, negative, and slightly negative. Each data point was annotated by three different people to obtain three different viewpoints, and the final tag was selected based on the majority's decisions. The dataset has an imbalanced distribution of data at each label. Figure 2 illustrates an example of the dataset. The frequency data for each sentiment label is shown in Fig. 3a.

As the amount of data at each label is small, we put positive, slightly positive classes in the positive category and negative, slightly negative classes in the negative category, keeping the neutral class as it was. Figure 3b shows the percentage of data at each newly annotated label.

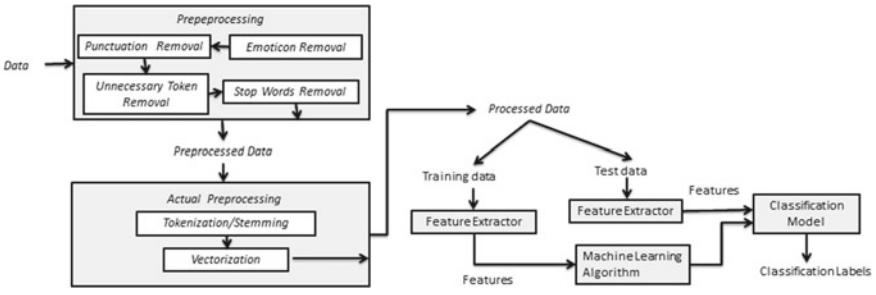

Fig. 1 System structure of the model

Comments	Sentiment
লিখার সময় পারলে সত্য লিখার অভ্যাস শিখুন।	কিছুটা নেতিবাচক
এটা কেন হচ্ছে? সংশ্লিষ্ট সকলের ডিপ্রেশনের ফলে? নাকি সরকার মনোনিত পরিচালনা পর্ষদের ব্যর্থতার কারণে?	কিছুটা নেতিবাচক
আমাদের দেশের স্বাভাবিক অর্থনৈতিক গতিপ্রবাহকে বাধাগ্রস্ত করা হয়েছে। তাই এসব ঘটছে।	নিশ্চিত নেতিবাচক

Fig. 2 Example of Bangla news comments with annotated sentiment label

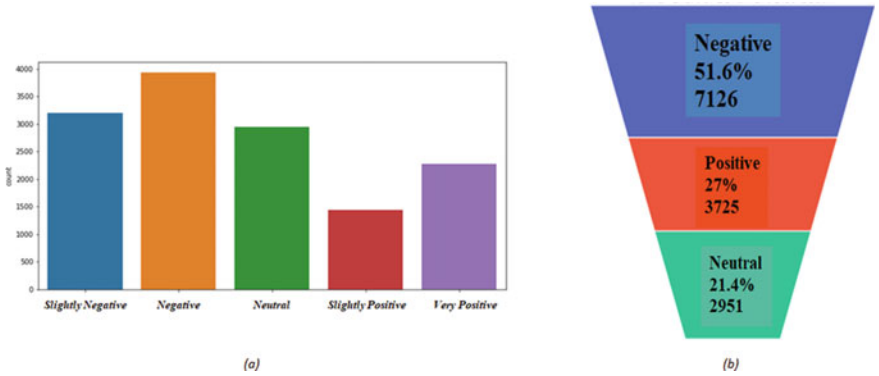

Fig. 3 **a** Amount of data at initial label, **b** percentage of data at newly annotated labels

3.2 Data Preprocessing

The initial raw data comprised of many redundant information and noise that did not offer anything to the sentiment analysis. To remove these, the data goes through several preprocessing steps. The preprocessing steps are given below:

(a) *The removal of emoticon*: Emoticons are basically used to represent different expressions and modes of the face. As in this paper, we are only working with textual information. We removed the emoticons, which do not contribute to our analysis.
(b) *The removal of punctuation marks*: Punctuation marks play a very minor role in determining the sentiment of a sentence. That is why punctuation marks are removed to avoid time and space complexity.
(c) *The removal of stop words*: Conjunctions and prepositions are common stop words. It is important to remove the stopwords to focus more on the important words and reduce complexity.

3.3 Actual Processing

A well-defined series of linguistically significant units was created from the preprocessed texts in this section. Tokenization or stemming of phrases, vectorization of tokens, and n-gram approaches are all involved in this process.

Tokenization and Stemming

In a sentence, a token consists of a sequence of characters that is regarded as a semantic unit. When a sentence is tokenized, the tokens are broken up into smaller chunks of information. For tokenization, we used whitespace as a delimiter. The technique of slicing words in order to return to their base words is known as stemming.

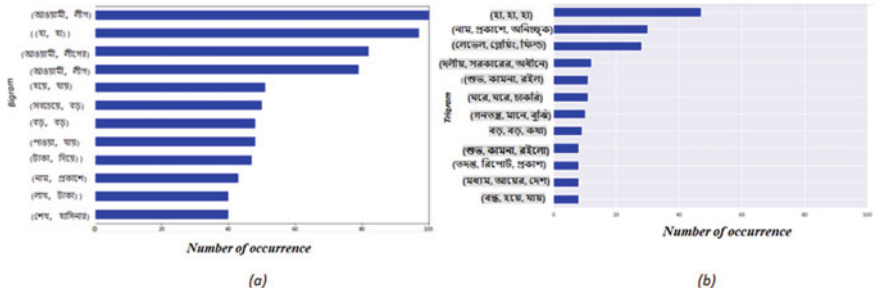

Fig. 4 Twelve most frequently occurring **a** bigram and **b** trigram of the dataset

Because each word derives its meaning from its root, removing affixes has no effect on opinion mining and simplifies calculation.

N-gram techniques We used our data to test various n-gram models in order to find the most useful feature. Based on $n-1$ prior words, n-gram anticipates the possibility of a term emerging. N-gram determines the associations between words and their neighbors and, to some extent, captures the context. Based on the value of n, the N-gram might be of numerous sorts. It is referred to as a unigram when n equals one. Only one word is used at a time by a unigram. The model is called a bigram for $n = 2$ and a trigram for $n = 3$. They, respectively, capture the context of two or three previous words. Figure 4 shows the 12 most frequently occurring bigrams and trigrams in our dataset.

3.4 Random Forest and Support Vector Machine

Random Forest Classifier

It enhances the projected accuracy of the dataset by averaging the results of many decision trees on different subsets of a dataset. It creates a classification category rather than a classification and then splits fresh data points based on the classifiers' predictions [13].

Support Vector Machine

SVM is used in machine learning algorithms, in which each node is represented as a data point in n-dimensional space, and every feature value relates to a specific coordinate. Then we classify the data by choosing the hyperplane that clearly divides the two groups [14]. To differentiate positive and negative data, SVMs look for the optimal surface.

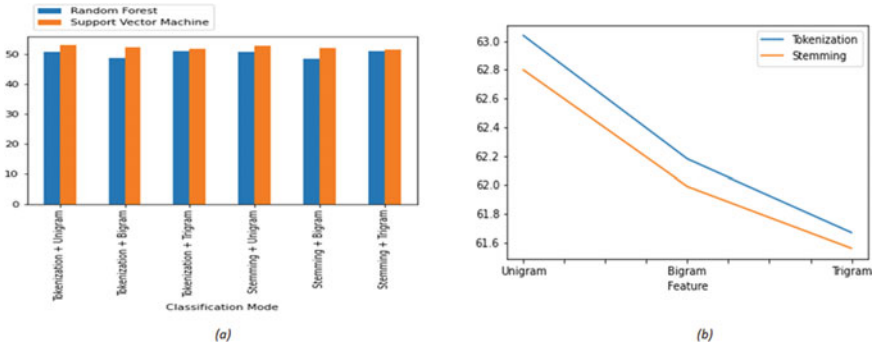

Fig. 5 **a** Comparison of SVM and random forest classifier, **b** comparison of n-grams in SVM

4 Result Analysis and Comparison

As our dataset is not properly balanced, we used four performance metrics to evaluate it accurately. These performance metrics are precision, recall, $F1$-measure, and accuracy. Better performance is associated with higher precision and recall values. Precision gives a measure of how precisely the system captures the correct cases, and recall gives a measure of the minimization of false negatives. The $F1$-measure is the reciprocal of the arithmetic mean of recall and precision. The equations for the performance metrics are given below, true positive and false positive are denoted by (TP) and (FP), true negative and false negative are denoted by (TN) and (FN), and m stands for sample size (TP + FP + FN + TN).

$$\text{Accuray} = \frac{\text{TP} + \text{TN}}{m} \quad \text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F1 \text{ measure} = \frac{(2 \times \text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Recall, $F1$-measure, and precision scores of each classifier's feature are all shown in Table 1, and the accuracy of the classifiers for each feature is shown in Table 2.

From the performance metrics, we can see that the classifiers predict negative opinions with high perfection (e.g., 96% recall using SVM) where neutral and positive classes are predicted with less perfection. We can say that the imbalance of labeled data has affected the performances highly. In both tokenization and stemming, SVM outperformed random forest. Tokenization has proven to be more effective for opinion mining than stemming. A comparison between the accuracy of SVM and random forest for each feature is given in Fig. 5a.

Table 1 Performance statistics of the proposed model

Features	Support vector machine				Random forest			
Tokenization + unigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	54	93	68	Neg.	54	85	66
	Neutral	22	2	3	Neutral	27	8	13
	Pos.	51	17	26	Pos.	43	20	27
Tokenization + bigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	53	94	68	Neg.	54	80	64
	Neutral	22	2	4	Neutral	21	8	12
	Pos.	49	13	20	Pos.	41	21	28
Tokenization + trigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	52	96	68	Neg.	53	92	67
	Neutral	15	1	2	Neutral	17	3	5
	Pos.	51	7	12	Pos.	45	10	17
Stemming + unigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	54	93	68	Neg.	54	84	66
	Neutral	23	2	3	Neutral	25	8	12
	Pos.	50	16	24	Pos.	44	21	28
Stemming + bigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	53	93	68	Neg.	53	80	64
	Neutral	25	2	4	Neutral	22	9	13
	Pos.	48	12	19	Pos.	39	21	27
Stemming + trigram		Precision (%)	Recall (%)	$F1$-measure (%)		Precision (%)	Recall (%)	$F1$-measure (%)
	Neg.	52	96	68	Neg.	53	92	67
	Neutral	15	1	2	Neutral	17	3	5
	Pos.	49	6	11	Pos.	44	10	17

We can see that unigram has shown better performance than bigram and trigram, though they capture more contextual information than unigram. We can say that the performance of n-gram depends highly on the characteristics of the elements of the dataset and their internal context and relationship among words. The comparison of n-gram techniques in the SVM classifier is shown in Fig. 5b.

Table 2 Performance scores of the proposed model

Accuracy		
Features	Support vector machine (%)	Random forest (%)
Tokenization + unigram	63.04	60.80
Tokenization + bigram	62.18	58.71
Tokenization + trigram	61.67	60.94
Stemming + unigram	62.80	60.80
Stemming + bigram	61.99	58.44
Stemming + trigram	61.56	60.92

5 Conclusion

SVM and RF are two of the most efficient machine learning algorithms in opinion mining that we used in conjunction with *n*-gram techniques to create a system for categorizing models for classification of Bangla text based on positive, negative, or neutral classes. Tokenization has showed greater results than stemming with a $F1$-score of 68%. SVM has shown better performance than the RF classifier and the best performance with unigram.

In the future, we want to introduce multi-classes in our system and expand the amount of data in our dataset. We also want to experiment with more preprocessing and feature selection techniques to improve the accuracy.

References

1. Tripathy A, Agrawal A (2015) Rath S (2015) Classification of sentimental reviews using machine learning techniques. Proc Comput Sci 57:821–829
2. Abu Taher SM, Akhter KA, Azharul Hasan KM (2018) N-gram based sentiment mining for bangla text using support vector machine. In: 2018 international conference on Bangla speech and language processing (ICBSLP). IEEE, Sept 2018
3. Ahammad K, Shawon JAB, Chakraborty P, Jahidul Islam M, Islam S (2021) Recognizing bengali sign language gestures for digits in real time using convolutional neural network. Int J Comput Sci Inf Secur (IJCSIS) 19(1):11–19
4. Chakraborty P, Ahmed S, Yousuf MA, Azad A, Alyami SA, Moni MA (2021) A human-robot interaction system calculating visual focus of human's attention level. IEEE Access 9:93409–93421
5. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up? sentiment classification using machine learning techniques. In: Proceedings of the 2002 conference on empirical methods in natural language processing (EMNLP 2002). Association for Computational Linguistics, pp 79–86
6. Pundlik S, Dasare P, Kasbekar P, Gawade A, Gaikwad G, Pundlik P (2016) Multiclass classification and class based sentiment analysis for Hindi language. In: 2016 international conference on advances in computing, communications and informatics (ICACCI). IEEE, pp 512–518
7. Rahman SSMM, Biplob KBMB, Rahman MH, Sarker K, Islam T (2020) An investigation and evaluation of n-gram, TF-IDF and ensemble methods in sentiment classification. In International conference on cyber security and computer science. Springer, pp 391–402

8. Tripto NI, Ali ME (2018) Detecting multilabel sentiment and emotions from bangla youtube comments. In: 2018 international conference on Bangla speech and language processing (ICBSLP). IEEE, pp 1–6
9. Rahman MM, Pramanik MA, Sadik R, Roy M, Chakraborty P (2021) Bangla documents classification using transformer based deep learning models. In: 2020 2nd international conference on sustainable technologies for industry 4.0 (STI). IEEE Publisher, pp 1–5
10. Rahman S, Chakraborty P (2021) Bangla document classification using deep recurrent neural network with bilstm. In: Proceedings of international conference on machine intelligence and data science applications. Springer, pp 507–519
11. Haque F, Manik MMH, Hashem MMA (2019) Opinion mining from Bangla and phonetic bangla reviews using vectorization methods. In: 2019 4th international conference on electrical information and communication technology (EICT). IEEE, pp 1–6
12. Akhter-Uz-Zaman Ashik M, Shovon S, Haque S (2019) Data set for sentiment analysis on Bengali news comments
13. Kominfo Makassar BBPSDMP (2019) Sentiment analysis using random forest algorithm-online social media based
14. Ashis P (2012) Support vector machine–a survey. Int J Emerging Technol Adv Eng 2(8):82–85