A Machine Learning Approach to Analyze Bengali Social Media Reviews

Ariful Hasan
Computer science and engineering
Daffodil International University
Dhaka, Bangladesh
email: Ariful15-10675@diu.edu.bd

Yeamin Mahmud
computer science and engineering
Daffodil International University
Dhaka, Bangladesh
email: yeamin15-10575@diu.edu.bd

Fahim Sakil Shuvo
computer science and engineering
Daffodil International University
Dhaka, Bangladesh
email: fahim15-10589@diu.edu.bd

***Abstract*—** this study presents a technique for categorizing Bengali serial reviews as good, negative, or spam. Document classification can be done in a variety of ways. This study provides a method for analyzing sentiment in Bengali serial reviews written in Bangla. This method may be used to analyze an audience's reaction to a certain film or television show. With an increasing number of individuals publicly sharing their opinions on social networking sites such as YouTube and Facebook, evaluating the sentiment of comments made about a certain movie or series might show how well the film is received by the general audience. This experiment's data was obtained and classified manually from publicly available social media comments. This model achieves 86.40 percent accuracy on the test set when using the KNN method, and 91.42 percent accuracy while using Random Forest. The model also achieves 93.3 percent accuracy by employing Logistic Regression. The model gets 92.14 percent accuracy when using Naive Bayes. Furthermore, a comparison with some other machine learning approaches is presented in this paper.

Keywords: Bengali sentiment analysis, Bengali Review Classifier, Naïve Bayes Algorithm, Machine Learning, Logistic Regression, Random Forest.

Introduction (*Heading 1*)

Bengali is the seventh most spoken language in the world and holds the second most spoken language in the Indian Subcontinent and the mother tongue of 'Bengali Nation'. Since the last decade use of the internet has been easier in this region. Hence many people started using social networking sites such as Twitter, Facebook, Youtube, etc. Youtube and Facebook are two of the most browsed sites in Bangladesh, thus media producers started publishing their serials, movies, tv-series on those platforms. It is easier to express opinions on social networking sites. Now Bangla lingual people can express their feelings in Bengali using Unicode keyboard apps such as Avro, ridmik, etc. They are easy to use and used by many. There are even smaller to larger online communities where people can discuss their feelings and share opinions. These opinions towards a series or movie can be positive or negative; some of the opinions or comments are totally irrelevant or margin away from the actual topic, we call them 'spam'. Analyzing such remarks can assist the producer in determining if the public views it positive or negative manner. Manually analyzing them (one by one) can be very time-consuming and inefficient too. As a result, this article examines the effectiveness of several machine learning classifiers that can quickly evaluate and categorize the Bangla comments.

Various machine learning models have been applied here to find the most efficient one. K-Nearest Neighbours (KNN), Random Forest Classifier, Nave Bayes Multinominal Classifier, and Logistic Regression were among the machine learning models used.

The rest of the article will be organized as follows: Sections II examines a related work, Section III briefly describes the dataset, Section IV examines the approach and system architecture, and Section V examines the dataset and preparation techniques utilized in this study. The experiment is described in Section V, and the findings are summarized in Section VI, which brings the study to a close.

ii. Related work

There has been a lot of sentiment analysis of series and movies in English, but there has been virtually no sentiment analysis of Bengali series or movies. Rumman Rashid Chowdhury, Mohammad Shahadat Hossain, Sazzad Hossain, and Karl Andersson's article, "Analyzing Sentiment of Movie Reviews in Bangla by Applying Machine Learning Techniques", analyzes the performance of SVM, Long-Short Time Memory [11],
and Naïve Bayes classification models. The dataset had 4000 samples, with 20% of the data being used for testing. SVM, Multinomial Nave Bayes, and LSTM were used to identify positives and negatives using natural language processing. SVM provided 87.02 percent accuracy, Multinomial Nave Bayes provided 88.38 percent accuracy, and LSTM provided 82.42 percent accuracy. The emphasis of this research is on document binary classification (positive and negative), for which SVM was found to be more efficient.

III. Dataset Preprocessing and Document Representation

People's remarks on social networking websites were manually gathered for this experiment. It contains around 13000 samples, each of which is carefully categorized as positive, negative, or spam. The remaining 20% of the data was utilized to create a test set, with the remaining 80% being used for training. The following are some examples from the dataset:

1 For true 0 for false

REVIEW	POSITIVE	NEGATIVE	SPAM
অনেক সুন্দর নাটক	1	0	0
অসাধারন একটা নাটক	1	0	0
তিশা,,এবং নিশু দুই জনকে অনেক সুন্দর লাগে।।	1	0	0
ভালো লাগলো না নাটক টা	0	1	0
সমাজটাকে শেষ করে দিল এই সব নাটক।	0	1	0
কি অশ্লীল ভংগী আর ভাষার ব্যাবহার	0	1	0

A. Preprocessing

On its own, the raw data obtained is insufficient for categorization. Several punctuation marks, emoticons, and other items in it are absurd to the sentiment analysis process. To improve accuracy, the dataset must be preprocessed before beginning the classification process. Depending on the language, a variety of pre-processing techniques are commonly used on data. Depending on the language, a variety of pre-processing techniques are commonly used on datasets. Preprocessing is an important step before beginning the classification process. The outcome of classification is determined by the success of preprocessing operations. Tokenization, punctuation and emoticon removal, stemming, and removal of stop-words are some of the procedures used.

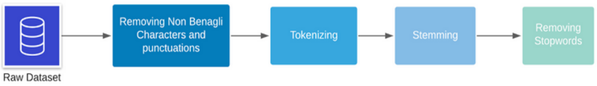

Figure 1. Preprocessing workflow

1. Removing Punctuations and other non-Bengali characters: Unwanted characters are removed during data preparation. Unnecessary components in the data were eliminated during the cleaning of non-Bengali characters on each data sample, such as punctuation marks, non-Bengali character alphabets, emojis, and so on.

2. Tokenization: Tokenization is the process of breaking down a text into logical tokens. Words, integers, and punctuation marks can all be used as tokens. Following that, an array of tokenized data sub-arrays was created. Each label was stored in its array. Here, whitespaces separated the words into tokens.

3) Stemming: Stemming is the process of reducing a word's variants to their most fundamental form. Depending on the situation, a word might take on a variety of distinct forms. For example, "করছে", "কর" is the base form for this term. The basic objective of stemming is to simplify the conjugational forms of a word to a single fundamental form. The overall amount of words that the classifier must deal with can be significantly reduced with this method. The widely common prefixes and postfixes used in Bangla were kept in an array to carry out this operation. A forked version of the 'Bangla-stemmer' package was used to identify the prefixes and postfixes, and the words' reduced versions were included in the newly processed corpus. Here “সত্যি অসাধারণ একটি রিলেশন”- upon stemming, the words (excluding stop words) look like this: [সত্যি, অসাধারণ, এক, রিলেশন] . Here, "একটি" has changed into its base "এক".

4) Stop word Removal: Stop words are words in a corpus of text that have little or no significance. When it comes to document classification, these concepts are meaningless. Stopwords in English contain phrases such as "so," "upon," "a," "an," and "the." Likewise, in Bangla, the phrases "অতএব", "এটা", "এটাই", etc. are thought of as stop words. Bangla stop word list was collected from ‘bltk’. After removing stop words from the phrase “সত্যি অসাধারণ একটি রিলেশন", the following tokens are obtained: [সত্যি, অসাধারণ, রিলেশন]. Because "একটি" was seen as a stop word in this context, it was omitted.

B. Document Representation

Representation of documents is a preprocessing approach for reducing the complexity of a dataset so that a model can handle it better. Existing texts must be converted to a representational vector. Vectorizer is a widely used document vector format in which documents are represented in word vectors. In this case, the Count Vectorizer was used for representing vectors and extracting features.

The CountVectorizer in Scikit-learn is used to transform a set of text documents into a vector of term/token counts. It also allows for the pre-processing of text input prior to vector representation generation. Because of its flexibility, it is a very adaptable feature representation module for text.

IV.METHODOLOGY AND SYSTEM ARCHITECTURE

Sentiment analysis may be performed using a variety of approaches. Depending on the dataset, the performance of each approach varies greatly.

Classic machine learning algorithms, such as KNN, Naive Bayes, Random Forest, and Logistic regression were used in our case.

Figure 2 depicts the workflow of the model building process.

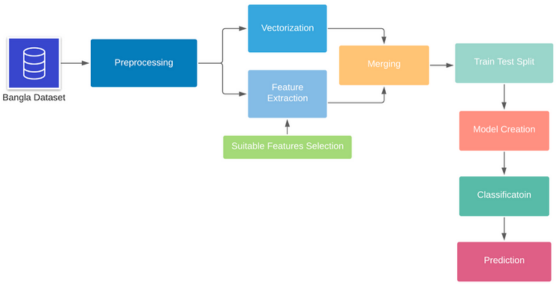

Figure 2. Model Construction workflow

A. KNN Algorithm

KNN is a non-parametric, slow learning technique. Its objective is to predict the classification of new sample points by utilizing a database containing data points separated into various groups.

In this research neighbor of K was 7 thus n_neighbors=7 was applied. To maximize the training process, we used n_jobs= -1 to use 100% of each logical core.

B. Random Forest Algorithm

A random forest is a machine learning technique for dealing with classification and regression problems. It employs ensemble learning, a technique for resolving complex problems by merging several classifiers. A random forest algorithm is made up of several decision trees. To maximize the training process, we used n_jobs= -1 to use 100% of each logical core.

C. Naive Bayes Algorithm

It's a probabilistic classifier that works on the assumption that the characteristics are unrelated. Many ML-related studies utilize it as a starting point.

D. Logistic Regression Algorithm

To learn from a dataset, the logistic regression model employs the sigmoid function. For implementation, the default Scikit-learn function is used.

V. *Experiment and Result Analysis*

A. The model achieved 91.42 percent accuracy on 20 percent data used as test dataset using Random Forest with Count Vectorizer, with 90 percent F1 Score, 91 percent Recall, and 91 percent Precision. Figure 3 shows a confusion matrix of the prediction findings. Table I displays information regarding the performance of the model.

Table I
Random Forest model performance metrics

Vectorizer	Accuracy	Recall	Precision	F1 Score
Count-Vectorizer	91.42%	91%	90%	90%

Confusion Matrix of Classifiers:

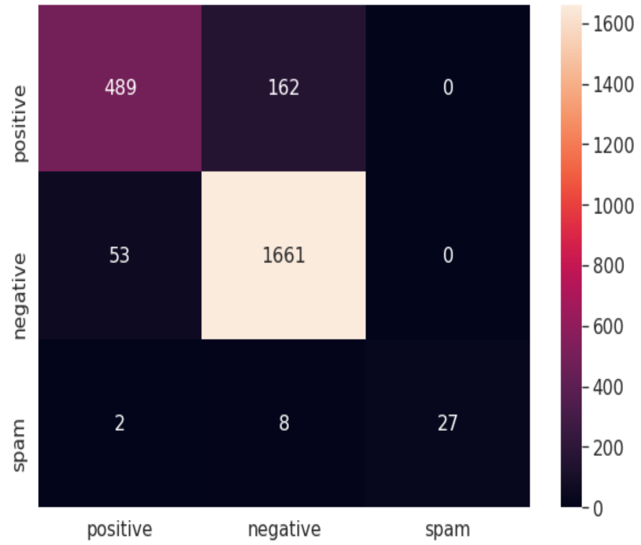

Fig: confusion matrix plot for Random Forest Classifier

B. The model achieved 86.40 percent accuracy on 20 percent data used as test dataset using KNN with Count Vectorizer, with 86 percent recall, 86 percent precision, and 85 percent F1 Score. Figure 3 shows a confusion matrix of the prediction findings. Table I displays information regarding the performance of the model.

Table I
KNN model Performance metrics

Vectorizer	Accuracy	Recall	Precision	F1 Score

Count-Vectorizer	86.4%	86%	86%	85%

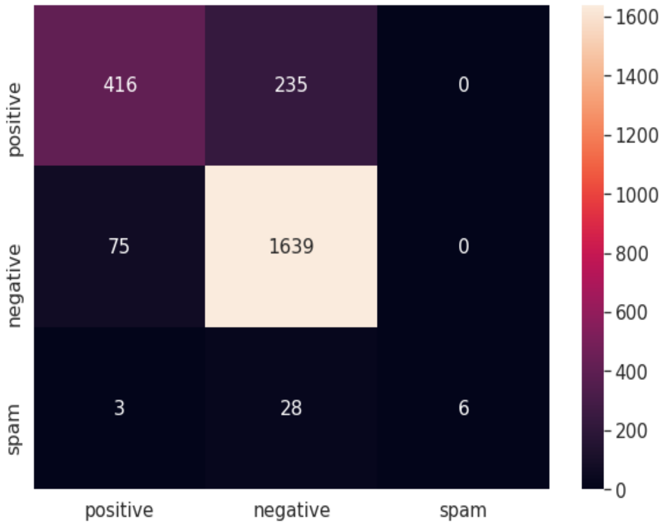

Fig: confusion matrix plot for KNN Classifier

C. The model achieved 92.14 percent accuracy on 20 percent data used as test dataset using Naive Bayes with Count Vectorizer, with 92 percent Recall, 92 percent Precision, and 91 percent F1 Score. Figure 3 shows a confusion matrix of the prediction findings. Table I displays information regarding the performance of the model.

Table I
Naive Bayes Classifier Performance metrics

Vectorizer	Accuracy	Recall	Precision	F1 Score
Count-Vectorizer	92.14%	92%	92%	91%

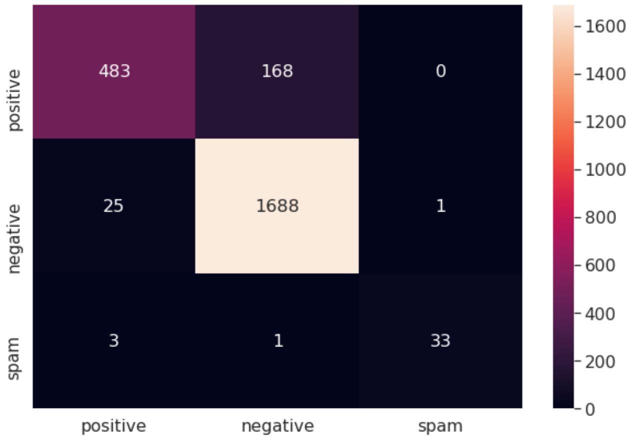

Fig: confusion matrix plot for Naive Bayes Classifier

D. The model achieved 93.3 percent accuracy on 20 percent data used as test dataset using Logistic Regression with Count Vectorizer, with 93 percent recall, 93 percent precision, and 92 percent F1 Score. Figure 3 shows a confusion matrix of the prediction findings. Table I displays information regarding the performance of the model.

Table I
Logistic Regression Performance metrics

Vectorizer	Accuracy	Recall	Precision	F1 Score
Count-Vectorizer	93.3%	93%	93%	92%

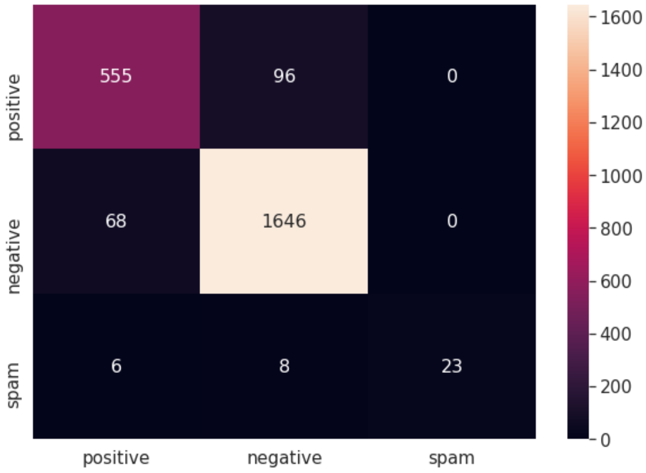

Fig: confusion matrix plot for Logistic Regression Classifier

VI. Conclusion and Future Work

This paper analyzes many strategies to utilize to analyze sentiment on Bangla movie review samples. Based on the results, it can be concluded that the Logistic Regression-based model outperformed the other approaches. We performed K-Fold Cross-validation on this model and got a satisfactory result there too. Because the number of “Spam” labeled samples in this experiment was tiny, the traditional models showed the most inaccuracy in them. However, the results may improve if samples for ‘spam’ are increased. Also, there are no good lemmatizer libraries for the Bengali Language. And the stemmers were also not very efficient and some of the lines of the codes had to be modified for this work. Even with so many limitations the models worked very well and can be used for accurate outcomes.

The table below shows a comparison of the approaches' performance

Table II
A comparison of the models' performance

Classifier	Accuracy
Random Forest	91.42%
KNN	86.40%
Multinomial Naive Bayes	92.14%
Logistic Regression	93.3%

The dataset must be expanded to include additional ‘Spam' data samples for this research to progress further. Grid search and any other extension might be utilized to increase the number of results.

REFERENCES

[1] Akhter, S. (2018, December). Social media bullying detection using machine learning on Bangla text. In 2018 10th International Conference on Electrical and Computer Engineering (ICECE) (pp. 385-388). IEEE.

[2] Chavan, G. S., Manjare, S., Hegde, P., & Sankhe, A. (2014). A survey of various machine learning techniques for text classification. International Journal of Engineering Trends and Technology (IJETT), 15(6), 288-292.

[3] Emon, E. A., Rahman, S., Banarjee, J., Das, A. K., & Mittra, T. (2019, June). A deep learning approach to detect abusive bengali text. In 2019 7th International Conference on Smart Computing & Communications (ICSCC) (pp. 1-5). IEEE.

[4] Eshan, S. C., & Hasan, M. S. (2017, December). An application of machine learning to detect abusive bengali text. In 2017 20th International Conference of Computer and Information Technology (ICCIT) (pp. 1-6). IEEE.

[5] Farhoodi, M., & Yari, A. (2010, November). Applying machine learning algorithms for automatic Persian text classification. In 2010 6th International Conference on Advanced Information Management and Service (IMS) (pp. 318-323). IEEE.

[6] Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. WSEAS transactions on computers, 4(8), 966-974.

[7] Kaur, J., & Saini, J. R. (2015). A study of text classification natural language processing algorithms for Indian languages. The VNSGU Journal of Science Technology, 4(1), 162-167.

[8] Miao, F., Zhang, P., Jin, L., & Wu, H. (2018, August). Chinese news text classification based on machine learning algorithm. In 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC) (Vol. 2, pp. 48-51). IEEE.

[9] Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep Learning--based Text Classification: A Comprehensive Review. ACM Computing Surveys (CSUR), 54(3), 1-40.

[10] Phani, S., Lahiri, S., & Biswas, A. (2016, November). A machine learning approach for authorship attribution for Bengali blogs. In 2016 International Conference on Asian Language Processing (IALP) (pp. 271-274). IEEE.

[11] R. R. Chowdhury, M. Shahadat Hossain, S. Hossain and K. Andersson, "Analyzing Sentiment of Movie Reviews in Bangla by Applying Machine Learning Techniques," *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)*, 2019, pp. 1-6, doi: 10.1109/ICBSLP47725.2019.201483.

[12] Razno, M. (2019). Machine learning text classification model with NLP approach. Computational Linguistics and Intelligent Systems, 2, 71-73.

[13] Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. Journal of machine learning research, 2(Nov), 45-66.

[14] Wang, Z. Q., Sun, X., Zhang, D. X., & Li, X. (2006, August). An optimal SVM-based text classification algorithm. In 2006 International Conference on Machine Learning and Cybernetics (pp. 1378-1381). IEEE.

[15] Zhang, W., & Gao, F. (2011). An improvement to naive bayes for text classification. Procedia Engineering, 15, 2160-2164.