4th International Conference on Electrical Information and Communication Technology (EICT), 20-22 December 2019, Khulna, Bangladesh

Opinion Mining from Bangla and Phonetic Bangla Reviews Using Vectorization Methods

Fabliha Haque, Md. Motaleb Hossen Manik and M.M.A. Hashem
Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
fablihahaque21992@gmail.com, mkmanik557@gmail.com, hashem@cse.kuet.ac.bd

Abstract— **Opinion mining is the computational study of people's opinions, emotions and attitudes which is one of the key research field in Natural Language Processing (NLP). To cope with the competitive world, owners of business need to extract exact opinion of people about his/her business. Recently, people in Bangladesh are more interested to express their opinion in Bangla and most importantly in Phonetic Bangla rather than English. Since no specific work of Opinion mining introduced this criteria, in this paper, we have developed review analysis system on Bangla and Phonetic Bangla where we have used Restaurant reviews as case study and the dataset is created manually by us without using translator. Our approach starts by preprocessing raw data and then feature extraction with different N-gram techniques. Then vectorization is applied on that data with HashingVectorizer, CountVectorizer and TF-IDF vectorizer. Later machine learning based approaches namely Support Vector Machine (SVM), Decision Tree (DT) and Logistic Regression (LR) are applied to classify reviews. We have classified the reviews in three different classes, i.e. bad, good and excellent. Finally a comparison is shown between vectorizers in accordance with different classifiers where SVM provides better accuracy with 75.58%.**

Keywords—Opinion Mining, Restaurant Reviews, Phonetic Bangla, HashingVectorizer, CountVectorizer, TF-IDF, Machine Learning

I. Introduction

Opinion Mining, also referred to as Sentiment Analysis, is an approach to natural language processing (NLP) that identifies the emotional tone behind a body of text. This is a popular way for organizations to determine and categorize opinions about a product, service or idea. This systems help organizations gather insights from unorganized and unstructured text that comes from online sources such as emails, blog posts, support tickets, web chats, social media channels, forums and comments. Businesses that use Opinion Mining tools can review customer feedback more regularly and proactively respond to changes of opinion within the market.Now-a-days, people visit different restaurants to avoid boredomness. In recent time, this is a trend that people check the reviews of restaurant to decide in which restaurant they should go. If customer choose a restaurant on the basis of few comments, then there may be a chance that the restaurant is not good actually. Again a restaurant authority always needs to analyze each and every customer review for business improvement. But in both cases, since the number of reviews are huge, it is not feasible to read these comments manually.

In today's world, total amount of internet user who has Bangla as native language has touched already 96.199 million till June 2019[1]. Among them, almost every people try to post at least one review after visiting a restaurant either in Bangla or in English or in **Phonetic Bangla** which is recent trend among smart phone users. These reviews can be either indicating that the restaurant is well enough to visit or below standard. Currently most of the restaurant has page in Facebook or related sites for reviews where people post their reviews. Again success of business relies on proper analysis of reviews. If restaurant authority cannot find out the problem faced by customers that is expressed through reviews, then it becomes tough for the authority to resolve the problems.

Since there are no rigid guidelines for posting comments and each day there are enormous amount of reviews posted, it is totally impossible to analyze reviews manually. So to develop an automated system for information gathering, Opinion Mining is a term which is the fundamental area of research in NLP [1]. This can determine the polarity of the review. This extracts attributes of the expression e.g. Polarity (if the speaker expresses a positive or negative opinion), Subject (the thing that is being talked about) and opinion holder (the person or entity that expresses the opinion) [2].

Currently available and ongoing researches have not addressed restaurant review analysis where reviews are in Bangla and most importantly in Phonetic Bangla. In our approach, initially the data is preprocessed including removal of punctuation, stopwords and other unnecessary symbols. For vectorization, we have tuned our model with HashingVectorizer, CountVectorizer and TF-IDF vectorizer to create the texts compatible for using machine learning techniques. Finally, for classification, we have preferred SVM which provides accuracy of 75.5% which is higher than any previously proposed model where they proposed their model only for Bangla and English comments. But we have simultaneously handled the reviews that are in Phonetic format like "**Ekhankar khabar onek valo legeche**" that implies "এখানকার খাবার অনেক ভাল লেগেছে" in actual Bangla. Even using different written format (Phonetic Bangla) of reviews, our model is providing better accuracy than any previous model. We have compared the accuracy with Decision tree and Logistic Regression where we have got accuracy level 66.67% and 73.18%, respectively.

The remaining paper is organized into four sections. **Section II** contains related works. **Section III** states our proposed model and methodology. Results are presented in **Section IV**. Finally, we have concluded and provided future directions in **Section V**.

II. Related Works

Bangla is a modified language which has complex structure of sentences and there is no proper NLP tools to perform research on Opinion Mining [3]. To resolve these restrictions researchers performed researches on Bangla language [4], they explained some automatic procedures to

[1] http://www.btrc.gov.bd/content/internet-subscribers-bangladesh-june-2019

978-1-7281-6040-5/19/$31.00 ©2019 IEEE

detect valence and emotion lexicons from text. In [5], they analyzed reviews of customers and identified polarity of reviews as positive, negative and neutral to assist a new customer through fuzzy logic model. But they are limited to small sized data and have not considered subjectivity and objectivity. Our inspiration mainly came from [6], where they performed Sentiment Analysis on multi-language (MSA, DA and English) dataset and finally applied Naïve Bayes and Decision Tree classifier. Sentiment classification techniques are applied to sentences in documentary to measure the polarity of movie reviews [7]. They have determined the scores using SentiWordNet dictionary [8] and given output through fuzzy logic approach. But their work is inappropriate for dataset having grammatically incorrect texts. A work on vectorization [9] is done where they have extracted emotions from text using TF-IDF for classification. They have used Support Vector Machine classifier and Vector Space Model (VSM) for representing document. Most of the above works have been done in English language rather than other native language. Rohini et al. [10] proposed a method for determining sentiment in an Indian native language i.e. Kannada. For classification they have used machine learning algorithms and finally deduced an accuracy level comparison between machine-translated English language and direct regional language (Kannada) dataset. But they failed to work with other Indian languages.

Animesh and Pintu [11], proposed a model where they performed feature selection process using Mutual Information [12] approach. They have applied Multinomial Naïve Bayes for classification and showed that Bangla dataset provides better accuracy than English using Bangla dataset from Amazon's Watches English dataset. Still they failed to process actual Bangla reviews. In report [13], Hidden Markov Model is used for POS tagging to determine polarity and then for classification, SVM is used. Rahman et al. [14] introduced Aspect Based Sentiment Analysis (ABSA) for the first time on Bangla text using two publicly available datasets of Cricket Review and Restaurant Review. Still they could not introduce actual Bangla or Phonetic Bangla from real Bengali reviewers.

Recently in [15], they did research in Bangla, English and Romanized Bangla comments fetched from YouTube. They proposed a deep learning based model for sentiment and emotion analysis where they used three class and five class sentiment and six class emotion labels. But their work failed to bring noticeable accuracy. Mahtab et al. [16], proposed machine learning based model where they worked on Bangla cricket commentary. They have used TF-IDF for vectorization and have used SVM as base classifier. But their work is limited to only Bangla comments and could not handle Phonetic Bangla. Again, they could not bring any noticeable accuracy on their own dataset.

Banik et el. [17] proposed two models on Bangla movie review classification where they have used SVM and Naïve Bayes for classification but could not come out with proper accuracy. Nidhi et el. [18] have surveyed on Indian native languages and have discussed different techniques (Machine learning, Lexicon Based and Hybrid) for SA. Al Amin et el. [19] have proposed an modified model of VADER for Bangla SA. But since it is modified form of English, so they could not bring desirable result. Most of the above works could not introduce Phonetic Bangla reviews classification and failed to bring proper accuracy for Sentiment Analysis.

III. Methodology

In this section, we have provided an overview of our proposed model. Initially, we have created dataset by collecting reviews from social media or sites and manually collecting from university students. We have completed many preprocessing steps to remove noise from raw data. Then tokenization is used for feature selection. Finally TF-IDF is used for vectorization and performance is evaluated by classifiers. Fig. 1 shows our system architecture.

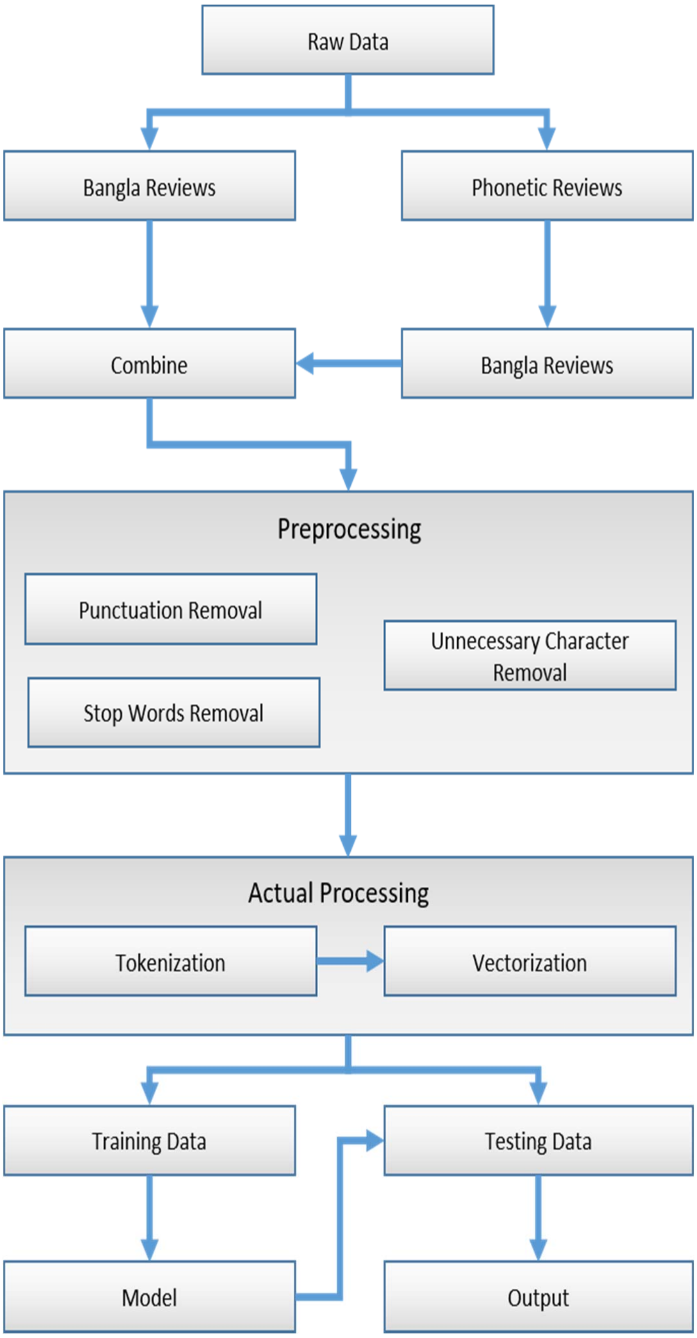

Fig. 1. Block Diagram of System Architecture

A. Dataset Creation

Since dataset on restaurant review in Bangla is not openly available, we have created our own dataset by collecting reviews from Facebook, YouTube videos, blogs and from university students with amount of 766, 174, 200 and 360, respectively. Our dataset contains reviews in both Bangla and

Phonetic Bangla. Though there was scope to collect reviews using different social media API, but it may contain noisy and un-structured data for our research purpose which may reduce accuracy. In our dataset, we have included two columns i.e. Review and Class. Each review falls in any class from **Bad**, **Good** and **Excellent** which is manually annotated. The shape of our dataset in 1500, including 500 bad, 500 good and 500 excellent reviews. Table I shows sample dataset.

TABLE I

SAMPLE CUSTOMER REVIEWS

Actual Review	**Original Form in Bangla**	**Class**
বিরিয়ানিটা একদম পারফেক্ট	বিরিয়ানিটা একদম পারফেক্ট	Excellent
Poriman aro barano dorkar	পরিমাণ আরও বাড়ানো দরকার	Bad
শিক্ষার্থীদের জন্য ভালো জায়গা	শিক্ষার্থীদের জন্য ভালো জায়গা	Good
Bargar ta onek valo lagse	বার্গারটা অনেক ভালো লাগছে	Excellent

B. *Reconstruction With Feasible Data*

In most of the previous works, opinion mining is done on Bangla text those are translated from English using translator that looks like reviews shown in Table II. But in most of the cases, these translation is not feasible. As in the first example, though translator provides a feasible Bangla text but in case of second example, it provides an infeasible Bangla text which contains complex words (**যুক্তিসঙ্গত, মহান**) that are not normally used in original review. People in Bangladesh normally express their feelings in easy text which is also our concern. Third column of Table II shows our approach of creating dataset with feasible text.

TABLE II

FEASIBLE REVIEW CREATION APPROACH

Review in English	**Review in Bangla through Translator**	**Our Approach**
The food was really good	খাবার সত্যিই ভাল ছিল	খাবার সত্যিই ভাল ছিল
Great food at reasonable price	যুক্তিসঙ্গত দামের মহান খাদ্য	ঠিক দামে ভালো খাবার
Interior was well decorated	অভ্যন্তর ভাল সাজানো ছিল	ভেতরটা অনেক ভালো ভাবে সাজানো ছিল

C. *Preprocessing*

Before normalization and analysis, a predictable or analyzable form is needed of texts because a text may contain many unnecessary words, punctuations regarded as noise which does not affect the polarity of the text. So we must remove these noisy parts of reviews for increasing the accuracy. A sample text of Bangla restaurant review which contains unnecessary words (stopwords), characters like punctuation and numbers is shown below.

এখানকার বার্গার অনেক ভালো।দামও কম মাত্র ৳১০০
লোকেশন@১২/এ বারিধারা

a) Punctuation removal: In any language, texts contain many punctuation characters that has less impact on sentiment analysis. Moreover they create the analyzing process complex. So to make process simpler, a list of Bangla punctuation characters (‘!’, ‘?’, ‘-‘, ’_’, ‘…’) is collected[2] and removed them from our dataset. These process has made our time complexity lesser than before.

b) Unncenessary characters removal: Raw reviews may contain many unnecessary characters *(* ‘=’, ‘<’, ‘>’, '%', '#', '@', '$', '*', '&') which are not required to be analysed for detecting the polarity of a particular review as these characters have no impact on the objectivity of the text. Rather keeping these characters may result in inaccurate analysis. They also require time to be processed. So, these characters are removed to normalize the data.

c) Stop words removal: There are some words in any language which appear frequently in text but convey no effect dealing with opinion mining. Some examples of these kind of words in Bangla language are ‘করি', 'যে', 'ছিলো', 'এখানে', 'কোনো', 'এখানকার', 'এবং'. As each word in text is considered as a feature in machine learning technique, so keeping stop words produce unnecessary features which do not affect the accuracy level rather increase complexity and also waste a lot of time for analysing. So, these stop words must need to be filtered which develops processing speed. To discard these type of words, we have created a list of Bangla stop words manually since no authentic Bangla stop words are available. A simple example from above example has been shown below, where all preprocessing mechanisms are used according to our model.

বার্গার অনেক ভালো দামও কম লোকেশন বারিধারা

D. *Actual Processing*

This section describes main processing steps that enable to continue toward classification and final comparison. Since we are working with Phonetic Bangla beside Bangla reviews, firstly we converted Phonetic Bangla reviews into actual Bangla. Reviews are now tokenized and later produced vectorized data. Then this data is used for feature selection and further process is done through classifiers. These steps are stated below:

a) Handling Phonetic Bangla Reviews: Normally smart phone users love to post review mostly in Phonetic Bangla rather than English. But they cannot be applied directly for processing. In our approach, we have converted Phonetic Bangla reviews into actual Bangla so that resulted dataset contains only Bangla reviews. Table III shows a sample of converted Bangla reviews from Phonetic Bangla through Python code. Algorithm 1 describes the procedure of implementing actual Bangla from Phonetic Bangla.

[2] http://www.grammarbd.com/en-grammar/punctuation

TABLE III

CONVERTED PHONETIC BANGLA COMMENTS

Actual Comment	Meaningful Comment
Ekane abr aste hobe	এখানে আবার আসতে হবে
Kishob ajebaje khabar	কি সব আজেবাজে খাবার
Ekhankar biriyan onek valo	এখানকার বিরিয়ানি অনেক ভালো
Poribeson valo legeche	পরিবেশন ভালো লেগেছে

Algorithm 1: Converting Phonetic Bangla to Bangla

```
Result: Bangla text
procedure PARSE (phonetic)
token ← each word from phonetic
declare empty array convertText
while token not null do
    word ← CONVERT (token)
    convertText ← addString (convertText, word)
return convertText
end procedure
```

b) Tokenization: Tokenization is taking a text or set of text and breaking it up into it individual words that bears a specific meaning [20]. For our model, we have used python library[3] to tokenize each review and considered them as feature. A complete tokenized version of previous example is showed below.

‘বার্গার’ ‘অনেক’ ‘ভালো’ ‘দামও’ ‘কম’ ‘লোকেশন’ ‘বারিধারা’

c) Feature Selection: There is always a change in result while tuning the feature. For our research purpose, we have used different N-gram ranges where N-gram are simply all combinations of adjacent words or letters of length n. The basic point of N-gram is that it captures the language structure from statistical point of view. Here N-gram range is the vital factor which depends on application as if N-gram is too short there may a chance to fail to capture important difference and if too long it may fail to capture the general knowledge. Based on our system, for better accuracy we applied both unigram and bigram. An English sentence “It looks very nice” is considered ‘It’, ‘looks’, ‘very’, ‘nice’ if Unigram is used. If Bigram is used then it will be considered as ‘It looks’, ‘looks very’, ‘very nice’.

In our approach we have applied unigram and bigram on Bangla and Phonetic Bangla which is shown in Fig. 2.

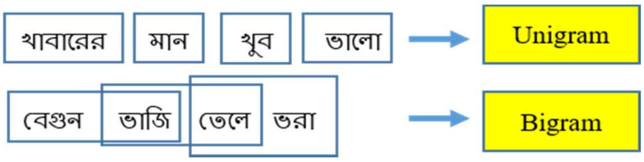

Fig. 2. Applying N-gram in our dataset

d) Applying Vectorization Methods: A machine cannot work with categorical data rather it only understands numerical data. So to feed machine a text corpus, categorical data needs to be converted into numerical value. This process refers to vectorization. Our initial approach was HashingVectorization which converts a collection of texts document into a sparse matrix holding token occurrance counts. This text vectorizer implementation usages hashing trick to find token stream name to feature integer index mapping.

N-gram or Bag-of-words produces sparse dataset containing n-1 features if the document contains n terms and this cause problem of curse of dimentionality and high memory consumption. To reudce the number of features we have applied **feature hashing** through HashingVectorizaiton where hash function is applied on each token. A feature number n is selected. Then remainder of each token’s hash value is generated by modulo n. These remainder value is mapped to the hash table where the token will be stored as 1. Table IV illustrates the feature hashing process on “খাবার বেশ মজার”.

TABLE IV

FEATURE HASHING

Token	Hash Value	Featuer number	Remainder
খাবার	836663229	5	4
বেশ	1007075880	5	0
মজার	1121767486	5	1

Table IV is mapped to Table V where index number is equal to the number of feature.

This vectorization technique was applied to our dataset but provided good result when the number of feature is huge (2^{20}). Since our dataset is not so large, so when we have applied this vectorization, then sometimes the hashing matrix was generated wider than the dictionary which cause many of the column entries in the hashing matrix as empty which is not acceptable in case of small document. Again, it may possible to reduce the length of hash feature which increases risk of collision where the function maps different term to the same feature that reduces the accuracy label. Moreover there is no way to inverse transform which can be a problem when trying to introspect which features are most important to a model.

TABLE V

SPARSE DATASET

Remainder	0	1	2	3	4
	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
খাবার	0	0	0	0	1
বেশ	1	0	0	0	0
মজার	0	1	0	0	0

We could have focused on **CountVectorizer** which creates the feature matrix with the frequency of terms. Though some common words (আমি, তুমি, সে, কবে, কিভাবে, দিকে, এবং, কিন্ত)

[3] https://www.nltk.org/api/nltk.tokenize.html

occur frequently in Bangla text but they convey very little impact on sentiment polarity. So, if we directly provide the numerical value of words with their frequency, then it will draw the performance of model down as less important terms get higher weight. So, re-weighting is needed to be given to feature.

So for this reasons we have applied Term Frequency Inverse Document Frequency (TF-IDF vectorizer) which is well performed with small dataset. TF-IDF scales term frequency counts in each document by penalising terms that appear more widely across the corpus. TF-IDF consist of two portion i.e. Term Frequency (TF), which is measured by,

$$\text{TF(t)} = \frac{Number\ of\ times\ term\ t\ appears\ in\ a\ document}{Total\ number\ of\ terms\ in\ the\ document} \qquad (1)$$

and Inverse Document Frequency (IDF), which is measured by

$$\text{IDF(t)= log(} \frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it}) \qquad (2)$$

TF-IDF simply deals with the relevance of word not with times of occurrences. By using this powerful vectorization method, we have got our desire feature vector ready for providing into machine.

e) Classification: Classification is an approach where new data is fallen in a class and this capability is given to machine by teaching it through using previously labelled data. There are many classifiers available for text classification. Among them Support Vector Machine, Decision Tree and Logistic Regression are most suitable for text classification. These are supervised algorithms performing better than other classifiers in case. In our dataset, most of the data are simply linearly separable. Since there are a large amount of feature to be considered in case of text, SVM with Linear Kernel performs better than other kernels i.e. RBF kernel, Polynomial Kernel. Moreover, we have also used Decision Tree and Logistic Regression classifier to analyse the comparative performance of our model.

IV. EXPERIMENTAL ANALYSIS

This section describes experimental evaluation of our proposed model on manually created dataset, where we have used three class label. Finally comparisons are shown through table.

A. Tuning Parameters

In our proposed model, we have used unigram and bigram as feature and for vectorization we have tuned through HashingVectorizer, CountVectorizer and TF-IDF vectorizer. These tuning generates different result. Again since our dataset contains 1500 comments in total with a combination on Bangla and converted Bangla through our Python code from Phonetic Bangla reviews, we have trained our model with equal size reviews of each type which is done by scikit-learn toolkit[4]. After that, unequal size review of each type is also trained to compare the result in both cases. We have primarily used 90% data for our training purpose and rest of the 10% data for testing. These training-testing size is also tuned to observe the result while it really shows effect on accuracy.

B. Result Evaluation

Result provided by any Machine Learning algorithm can is evaluated by some terms i.e. Accuracy, Precision, Recall, F1-Score. While considering these terms, four new terms arise i.e. True Positive, False Positive, False Negative and True Negative. If the sample size in N, then a simple equation can define overall accuracy of a system by

$$\text{Accuracy} = \frac{True\ Positive+True\ Negative}{N} \qquad (3)$$

$$\text{Precision} = \frac{True\ Positive}{True\ Positive+False\ Positive} \qquad (4)$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive+False\ Negavite} \qquad (5)$$

$$\text{F1-Score} = \frac{2*Precision*Recall}{Precision+Recall} \qquad (6)$$

Table VI shows performance evaluation of our proposed model where different features and review types (without Phonetic Bangla and with Phonetic Bangla) are tuned to observe the results.

TABLE VI

PERFORMANCE EVALUATION

Features	**Precission**	**Recall**	**F1-Score**
Unigram+SVM	0.80	0.79	0.79
Unigram+DT	0.68	0.64	0.64
Unigram+LR	0.67	0.62	0.62
Unigram+SVM+Phonetic	0.75	0.79	0.79
Unigram+DT+Phonetic	0.65	0.65	0.65
Unigram+LR+Phonetic	0.73	0.72	0.72
Bigram+SVM	0.76	0.75	0.74
Bigram +DT	0.70	0.65	0.65
Bigram +LR	0.76	0.75	0.75
Bigram +SVM+Phonetic	0.76	0.75	0.74
Bigram +DT+Phonetic	0.67	0.68	0.68
Bigram +LR+Phonetic	0.73	0.74	0.74

From our ovservation we have found that SVM providing more accurate result than other classifiers with parameter tuning. In some cases, LR shows equal result of SVM. From Fig. 3, we can see that SVM performs better. In our model most of the time classifiers predict ‘Bad’ reviews accurately. But since ‘Good’ and ‘Excellent’ reviews contain almost same type of words, that is why for some cases classifiers confused to differentiate between ‘Good’ and ‘Excellent’ reviews. These reviews are so confusing that even human may misclassify them. Since we have applied different N-gram and vectorization methods, our model provides different accuracy for different combination of N-gram and Vectoriztion methods. Table VII provides a comparative analysis of these combinations.

Finally Fig. 4 shows a comparison of accuracy with different vectorizer on SVM where it is noticable that SVM

[4] https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

shows equal accuracy everytime but in HashingVectorizer if the number of feature is increased than accuracy increases.

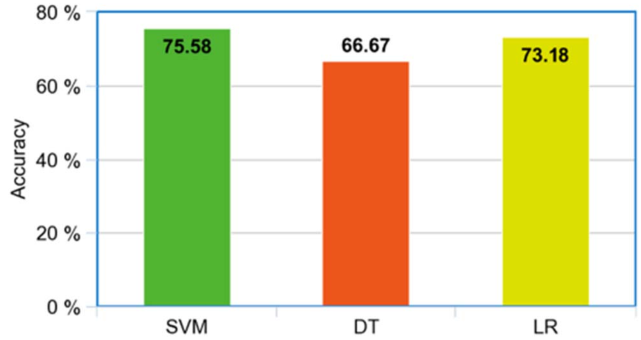

Fig. 3. Comparison between different classifier's accuracy

TABLE VII

COMPARISON OF ACCURACY WITH DIFFERENT N-GRAM AND VECTORIZER

Feature	SVM	DT	LR
HashingVectorizer + Unigram	75.0	62.04	66.67
HashingVectorizer + Bigram	74.07	61.11	73.15
CountVectorizer + Unigram	75.00	69.44	67.59
CountVectorizer + Bigram	73.15	61.11	68.52
TF-IDF Vectorizer + Unigram	73.00	67.59	64.81
TF-IDF Vectorizer + Bigram	75.58	66.67	73.18

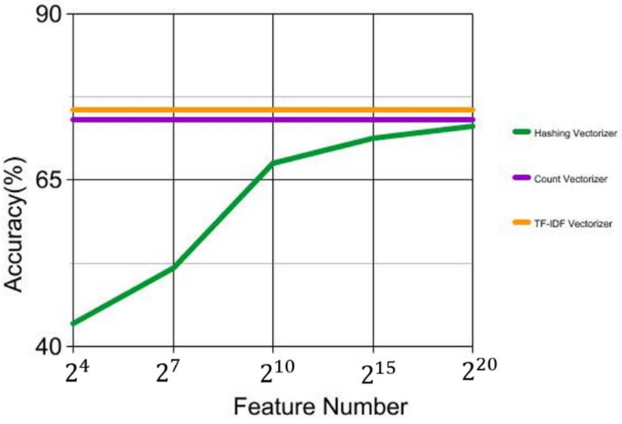

Fig. 4. Accuracy comparison with different vectorizer

V. CONCLUSION

In our paper, we have introduced a model that works on manually annotated restaurant reviews. Since no open dataset is available in this domain, we have collected reviews from different social media and by performing a survey on university students where the reviews are in both Bangla and Phonetic Bangla. Then these raw data are preprocessed. Then this preprocessed data is reformed through different vectorization methods for making them applicable to machine learning techniques (SVM, DT, and LR). Since our dataset is small, we have not got desire accuracy level like English. However our model provides better accuracy than previously developed models where they did not handle Phonetic Bangla. In future, we will enlarge our dataset and will add more classes. Moreover, we will add new class label where it will define either a review is about food or about behaviour of staffs or about the restaurant's environment. We will also design an unsupervised model for classifying reviews.

REFERENCE

[1] https://towardsdatascience.com/real-time-sentiment-analysis-onsocial-media-with-open-source-tools-f864ca239afe (accessed on:15.09.2019)

[2] https://towardsdatascience.com/deep-learning-for-sentimentanalysis7da8006bf6c1 (accessed on:16.09.2019)

[3] M. Karim, Technical challenges and design issues in bangla language, IGI Global, 2013

[4] S. M. Mohammad. Sentiment analysis: Detecting valence, emotions and other affectual states from text. In Emotion measurement, pages 201–237. Elsevier, 2016

[5] J. D. Silva and P. S. Haddela. A term weighting method for Identifying emotions from text content. In Industrial and Information Systems (ICIIS), 2013 8th IEEE International Conference on, pages 381–386. IEEE, 2013

[6] M.E.M. Abo et al., "Sentiment analysis algorithms: evaluation performance of the Arabic and English language", 2018 International Conference on Computer Control Electrical and Electronics Engineering (ICCCEEE), 2018.

[7] Pranali Tumsare, Ashish S Sambare, Sachin R Jain, and Andrada Olah. Opinion mining in natural language processing using sentiwordnet and fuzzy. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) Volume, 3:154–158, 2014.

[8] S. Baccianella, A. Esuli, and F. Sebastiani,"Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining." In LREC, vol. 10, 2010, pp. 2200–2204.

[9] J. D. Silva and P. S. Haddela. A term weighting method for identifying emotions from text content. In Industrial and Information Systems (ICIIS), 2013 8th IEEE International Conference on, pages 381–386. IEEE, 2013.

[10] V. Rohini, M. Thomas, C. Latha, "Domain based sentiment analysis in regional language-Kannada using machine learning algorithm", 2016 IEEE International Conference on Recent Trends in Electronics Information & Communication Technology (RTEICT), 2016.

[11] A. K. Paul and P. C. Shill, "Sentiment mining from bangla data using mutual information," in Electrical, Computer & Telecommunication Engineering (ICECTE), International Conference on. IEEE, 2016, pp. 1–4.

[12] H. Liu, J. Sun, L. Liu, and H. Zhang, "Feature selection with dynamic mutual information," Pattern Recognition, vol. 42, no. 7, pp. 1330–1339, 2009.

[13] A. Roy and A. A. Singh. (2018, Oct.) Sentiment Analysis ANLP Research Report.

[14] M. A. Rahman and E. Kumar Dey, "Datasets for aspect-based sentiment analysis in bangla and its baseline evaluation," Data, vol. 3, no. 2, 2018.

[15] N.I. Tripto and M.E. Ali, "Detecting Multilabel Sentiment and Emotions from Bangla YouTube Comments", 2018 International Conference on Speech and Language Processing (ICBSLP), 2018.

[16] S.A. Mahtab et al.," Sentiment Analysis on Bangladesh Cricket with Support Vector Machine" 2018 International Conference on Speech and Language Processing (ICBSLP), 2018.

[17] N. Banik , H. Rahman, "Evaluation of Naïve Bayes and Support Vector Machines on Bangla Textual Movie Reviews," International Conference on Bangla Speech and Language Processing(ICBSLP) on IEEE,2018 pp. 1-6

[18] N. Hadia, N. Nanavita, "Indic SentiReview: Natural Language Processing based Sentiment Analysis on major Indian Languages," Proceedings of the Third International Conference on Computing Methodologies and Communication (ICCMC 2019), on IEEE Xplore, 2019, pp.1-6

[19] A. Amin et el, "Bengali VADER: A Sentiment Analysis Approach Using Modified VADER," 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), 2019, pp. 1-6

[20] http://blog.kaggle.com/2017/08/25/data-science-101-getting-started-in-nlp-tokenization-tutorial (accessed on:19.09.2019)