# A Comparative Study on Bengali Speech Sentiment Analysis Based on Audio Data

Abanti Chakraborty Shruti
*Dept. of Computer Science and Engineering*
*BRAC University*
Mohakhali, Dhaka-1212, Bangladesh
abanti.chakraborty.shruti@g.bracu.ac.bd

Rakib Hossain Rifat
*Dept. of Computer Science and Engineering*
*BRAC University*
Mohakhali, Dhaka-1212, Bangladesh
rakib.hossain.rifat@g.bracu.ac.bd

Marufa Kamal
*Dept. of Computer Science and Engineering*
*BRAC University*
Mohakhali, Dhaka-1212, Bangladesh
marufa.kamal1@g.bracu.ac.bd

Md. Golam Rabiul Alam
*Dept. of Computer Science and Engineering*
*BRAC University*
Mohakhali, Dhaka-1212, Bangladesh
rabiul.alam@bracu.ac.bd

***Abstract*—Sentiment analysis is one of the most researched areas for every language. Due to the rise of AI, the use of speech in every sector is rapidly growing so is the importance of Speech Sentiment Analysis. Despite being the seventh most spoken language in the world, Bengali speech sentiment analysis studies are not much enriched. This study compared the Bengali speech sentiment analysis using machine learning and CNN, LSTM, and Bi-LSTM models. We have used the SUBESCO and BanglaSER datasets for training our models where the KNN model outperformed other models with an accuracy of 90%. Later, we evaluated the performance of the models with our custom-made test dataset. Experimental results show that AdaBoost and Bi-LSTM model performed best with 45% accuracy. Moreover, to understand the feature effect on the output, we used the interpretable SHAP model in the ML model outcomes as they provide the best results allowing us to have an explainable advantage to determine the results.**

***Index Terms*—Bangla Sentiment Analysis, Machine Learning, CNN, SHAP, MFCC, KNN, AdaBoost, Random Forest, Explanable AI, LSTM, Bi-LSTM**

## I. Introduction

Sentiment analysis is the study of how individuals feel or behave in response to an occasion, a topic of discourse, or in general using text analysis, audio speech, biometrics, and other features which may provide clear distinction in the emotions that are portrayed through the data received. Sentiment Analysis, shortly referred to as SA is making its contribution in various sectors such as identifying the sentiment of a product review [1] [2] which in turn helps a business to make decisions, SA in education [3] to improve the quality of teaching for students, development of artificial intelligence [4] to understand human emotions, etc. The majority of the work in sentiment analysis over the last few decades has focused on textual sentiment analysis using text mining techniques [5] leaving a large scope for audio data specially for low languages such as Bengali. It is generally associated with NLP which determines whether a text is positive, negative, or neutral based on the context of the sentence or paragraph [6].

In comparison to textual data interpretation, audio speech analysis has been deprived of its fair share of research despite the fact that audio data may elicit a wide range of human emotions and may result in more insights.

Research on low language resources such as the Bengali language still needs a lot of attention considering there are about 210 million who speak Bengali, and sentiment analysis on audio data has not gained much traction as of yet. This research paper approaches a comparative study of multi-class classification of sentiment in Bengali language audio speech by extracting the emotions using different feature extraction methods including the Mel frequency cepstrum coefficient (MFCC) feature extraction technique. This technique has become a state-of-the-art for speech recognition model after its introduction by Davis and Mermelstein in 1980 [7]. It works well in the case of speech recognition systems as its performance is the signal-to-noise ratio. Additionally, we also experimented with other feature extraction methods such as Chroma Shift, Zero Crossing Rate (ZCR), and Root Mean Square(RMS) and even a combination of them to evaluate the differences. Furthermore, two Bengali corpora have been combined and used to investigate the result. The first one is a public dataset - SUST Bangla Emotional speech corpus (SUBESCO) [8] and the second one is BanglaSER: A speech emotion recognition dataset for the Bangla language [9]. These datasets contain speaker-specific audio(.wav) files with different emotions. Since the audio files were recorded without any noise, we have also augmented the data with noise additionally adding pitch and stretch to evaluate the sentiment analysis of positive, negative, and neutral classes more efficiently on real audio data.

Although deep learning approaches such as DNN, RNN, Bi-LSTM, autoencoders, and attention mechanisms are popular for sentiment analysis [10], we have experimented with novel machine learning algorithms for classification along with CNN pre-trained models and LSTM and Bi-LSTM networks. A

comparison of the highest performance based on K-nearest neighbor, AdaBoost, Random forest classifiers, CNN models, LSTM, and Bi-LSTM model have been demonstrated in our research which is tested on our validation dataset as well as our custom human annotated test dataset. In summary, the key objectives of the research are as follows:

- Preprocess the data and extract features to find correlations between the same sentiment.
- Preparing the image dataset of Mel spectrogram from audio signals for the three different sentiments
- Compare between Random Forest, KNN, AdaBoost, CNN, LSTM, Bi-LSTM models to find out the highest performing classification technique.
- Prepare human annotated real-life audio dataset consisting of 150 files to test the best model classifier.
- Using different evaluation metrics: Accuracy, Precision, Recall, and F1-Score, we evaluate the performance of our classifiers. Also, evaluate the effectiveness of the model by presenting the confusion matrix.
- Explainable AI SHAP models were used which give us an understanding of the entire model and prediction of the genres more precisely.

The remainder of the study is arranged as follows: Section II discusses the related papers on sentiment analysis and emotion classification based on English and other languages. Section III discussed and outlined the research methodology. Section IV shows the datasets used and prepared. Section V displays the analysis of the results. Finally, in Section VI, the conclusion and future scope of the research are provided.

## II. Related Works

Speech Sentiment Analysis (SSA) study is a rapidly growing research sector due to the increasing use of audio and voice in everyday life scenarios. Numerous studies have been done to increase the efficiency of SSA in different languages, especially in English. Like all other languages, the SSA in Bangla language is also an important study as approximately 210 million people speak Bangla. However, there are very few studies covering the SSA for the Bangla language among which the study in [8] is a prominent one that proposed a Speech Emotion Recognition(SER) architecture named DCTFB that combines the Deep CNN and Bi-LSTM with a TDF layer. For the experiments, the audio-only Bangla emotional speech dataset SUBESCO, and the RAVDESS are used. The model can acquire both local and sequential information about emotional speech. They experimented with 8 models including the DCTFB model which achieved the best results: a weighted average(WA) accuracy of 86.86% with an average f1 score of 86.86% for the SUBESCO dataset. For the RAVDESS dataset, the model acquired a WA accuracy of 82.7%. Another research work [11] proposed a system that recognizes the emotion from Bengali speech. MFCC and LPC are combined to extract features from the speech signal. Multiple machine learning algorithms like SVM, KNN, AdaBoost, Logistic Regression, and XGBoost are used in this system to predict the sentiment among which LR and SVM performed best. A self-collected audio recording dataset named Abeg containing 301 audio speeches and the popular RAVDESS dataset is used to measure the performance of the system. The logistic regression model achieved the highest accuracy of 92% on the Abeg dataset. When Abeg and RAVDESS datasets are combined the XGBoost classifier achieved an accuracy of 86% for twelve users. An automatic speech recognition system that identifies isolated Bangali words is proposed in the paper [12] which achieved 86.08% accuracy. SVM with dynamic time wrapping(DTW) has been used in this model. MFCC is used to identify the features of audio while DTW is used for feature matching. Later, using these features SVM is used for classification in this model. For training, the study used a self-collected dataset of five Bangla words that are spoken by 40 different speakers. A novel framework for performing sentiment analysis on word-based Urdu speech was proposed in [13] where, short-term audio features are extracted using MFCC, PLP, Spectral energy, and Chroma vector features, and then five mid-term features are processed from them. These mid-term features are then used as input to find out the sentiment of Urdu utterances. HMM and DTW is fused to get the final opinion. To evaluate the model, an Urdu custom corpus of 600 words is used which achieves 97.1% accuracy.

Most of the studies are using deep learning-based models to increase the performance of the models. A deep learning-based model to increase the accuracy and prediction rate of the existing models is proposed in the study [14]. The main focus of this study was to improve the extraction of speech feature information. A new framework adopting a hierarchical conformal model to extract the spatiotemporal features of the speech and attention-based GRU model to fusion the features is presented in this work. The model also focused on reducing the computational cost of the feature extraction deep learning model. IEMOCAP and RAVDESS benchmark dataset is used to evaluate the performance of the new framework which shows 80% and 81% of accuracy respectively and outperforms the existing models. In the paper [15], the focus is on the heterogeneous audio signal features depending on which audio sentiment analysis is performed. An utterance-based deep neural network model combining CNN and LSTM is proposed in this study to identify the features of audio. MFCC and other recognized feature extraction methods are also used to find out homogenous features. Further, attention-based Bi-LSTM has been used to fusion the features. The MOSI dataset has been used to train the model and the Spanish dataset MOUD is used for testing purposes. The model has a better performance of 9.33% than the state-of-art models.

## III. METHODOLOGY

In this paper, we experiment with different machine learning(ML) and CNN, LSTM, Bi-LSTM models which use features extracted from the augmented audio signal data to detect the correct sentiment of the audio. The process involved different steps which are briefly discussed as follows:

### A. Data Augmentation

For improving the generalization capability of our models we first used data augmentation so that the data becomes syntactic data. As we can see both of our datasets [1][2] contains audio which are recorded in a controlled environment keeping many parameters in mind. In real life, the data is not always so clean and has no environmental constraints. To make the data more general to the real-life data we added noise, pitch, and stretch on the raw audio of our training dataset.

For augmentation, we used the **librosa** library to add the attributes. First, we injected some random noise into our raw data. Then we added pitch factor 0.7 by randomly modifying the frequency of the noisy data followed by adding stretch factor 0.8 which increased the time by slowing down the data.

### B. Feature extraction

After the data has been augmented with noise, pitch, and stretch, to extract the features from the augmented data we experimented with different feature extraction techniques. Humans have a nonlinear scale for auditory perception, MFCC tries to replicate the human ear as a mathematical model. A signal's **Mel frequency cepstral coefficients (MFCCs)** are a limited group of characteristics (typically 10-20) that simply defines the overall shape of a spectralenvelope [16]. *librosa.feature.mfcc* is used to compute the pandas data frame of our training dataset consisting of the 20 features. **Zero-crossing rate(ZCR)** is a frequency which is the rate at which the sign of signal changes from negative to positive or vice versa and produces a frequency in between. Using *librosa.zero crossings()* function the zero crossing for each frame is calculated. Another extraction technique **Chroma Shift** has been used to evaluate the tonal differences of the audio signals represented in a condensed form. *librosa.feature.chroma_stft()* function helps to find the harmonic similarity and detect chords of the audio [17]. Lastly, we have used **Root Mean Square(RMS)** using *librosa.feature.rms()* function to extract the features from the audio files.

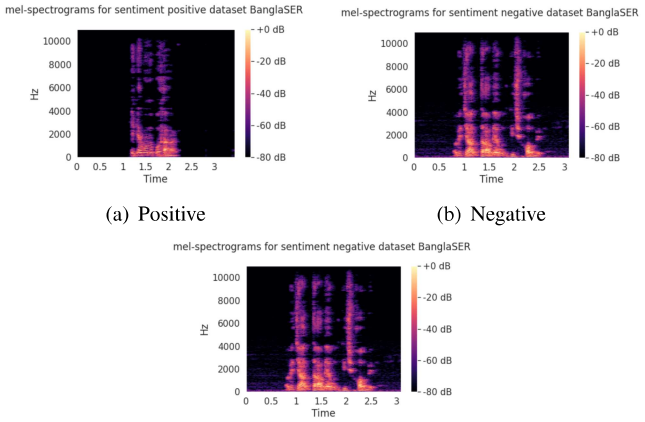
### C. Re sampling and Normalization

Re-Sampling: As our dataset was a merged dataset of two different datasets it had an imbalance amount of data in the three classes. That is why we used oversampling method to balance the dataset. SMOTE from the library imblearn.over_sampling has been used to resample the dataset.

Normalization: Scaling has been done to normalize our data. StandardScaler() has been used to transform our dataset and perform normalization.

### D. Mel Spectogram Generation

For our CNN models, we have generated the Mel-spectrograms from each of the audio files. Using the Mel spectrogram method of librosa library we transformed each and every audio file of our training dataset and test dataset. We saved every spectrogram as an individual png file and then appended the class labels with them creating a CSV dataset file. These spectrograms were further used in our CNN models to train and predict the sentiments.




(a) Positive (b) Negative

(c) Neutral

Fig. 1. Mel Spectograms of Three Sentiment Classes

### E. Sentiment Analysis Based on Machine Learning Models

The variety of categorization techniques makes it difficult to choose just one single classification technique. The focus of this study lies on 3 different types of supervised machine learning models that will classify the augmented data displaying the best result.

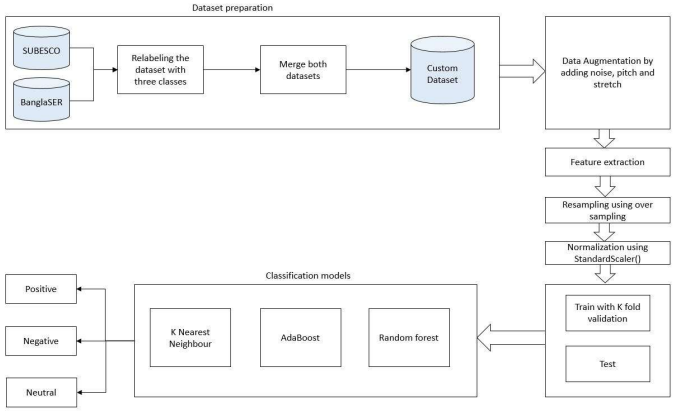



Fig. 2. Proposed Machine Learning Model Architecture

*1)* ***Random Forest (RF)****:* The random forest also known as random decision forest, is a supervised machine learning model which uses an ensemble of learning methods for classification, regression, and other tasks. It uses many decision trees on different subsets of the input dataset and averages the results to increase the dataset's predicted accuracy. Random subsets are created from the input dataset. Gini impurity helps to construct a decision tree for each data subset. If the sum of the GINI impurity of the split sub-tree is lower than the GINI impurity of the parent node then the parent node is split further. Finally, based on the majority of votes, a bagging technique selects the final output.

$$I = G_{\text{parent}} - G_{\text{split1}} - G_{\text{split2}} \tag{1}$$

$$G = \sum_{i=1}^{C} p_i * (1 - p_i) \tag{2}$$

Here, summation occurs for all classes C using the probabilities. G stands for the Gini impurity and I for the intensity. We have used k = 5 folds for cross-validation, 50 random decision trees, and gini impurity which finally resulted in one leaf node each.

*2) **K-Nearest Neighbour(KNN)**:* KNN is one of the simplest supervised machine learning techniques which is widely used for classification predictions. This algorithm groups the data based on their similarities. The KNN model takes a data point with an unknown class and finds its similarity with all other data points of the dataset and then chooses the K number of nearest neighbors with the most similarity. From the KN neighbors, it takes the majority class votes and decides the class of the unknown data. In our model, we have used the "Euclidean" distance to find out the similarities of our data and K=5 for cross-validation, and 6 nearest neighbors are taken into account for majority voting.

$$EuclideanDistance(x, y) = \sqrt{\sum_{i=1}^{n} (x_i^2 - y_i^2)^2} \tag{3}$$

where,

$(x_i, y_i)$ = x,y coordinates of data points
$n$ = total number of data points

*3) **AdaBoost**:* AdaBoost(Adaptive Boosting) is an ensemble learning model which was presented to improve the power of prediction of weak learners. It combines the mistakes of weak learners and designs a strong learner to solve binary classification problems. AdaBoost works by building a number of decision stumps by dividing and splitting the examples into two subsets of one feature and predicting the output based on them and using GINI impurity to calculate the decisions. These are called weak learners as they only work on one feature. But in a real scenario, a decision depends on multiple features that are where ensemble learning comes into the picture. First, all the examples have the same weight.

$$w = 1/N \tag{4}$$

where,

$w$ = weight of the data point
$N$ = total number of data points

The adaBoost algorithm then learns from the mistakes of the predecessor weak learners and creates a new decision stump with more weight on the wrong classified features until data points are correctly predicted. Decision Stumps are like trees in a Random Forest, but not "fully grown." They have one node and two leaves. AdaBoost uses a forest of such stumps rather than trees. The influence of classifiers is calculated as follows -

$$\alpha_t = \frac{1}{2} ln \frac{(1 - totalError)}{totalError} \tag{5}$$

where,

$\alpha$ = the amount of influence the stump has in the classification

$$\text{totalError} = \frac{\text{total number of misclassified data points}}{\text{total number of data points}} \tag{6}$$

Using the alpha the weight value is updated as follows-

$$w_i = w_{i-1} * e^{\pm\alpha} \tag{7}$$

Here, alpha is positive when data is correctly classified and is negative otherwise. In our model, we have used a 5-fold validation for 250 estimators with a learning rate of 0.1.

### *F. Sentiment Analysis Based on Deep Neural Network Approaches(DNN)*

**Convolutional Neural Network(CNN)**

Computer vision methods for deep learning models are the most popular sector of studies. Generally, CNN's are made up of multiple convolutions layers and fully connected layers and is hugely used for classification problems. In this study, we have focused on using the CNN models to classify the sentiments of the speech by transforming the audio signal to spectrogram images. We have used a total of three pre-trained network models which are described in the next sections.

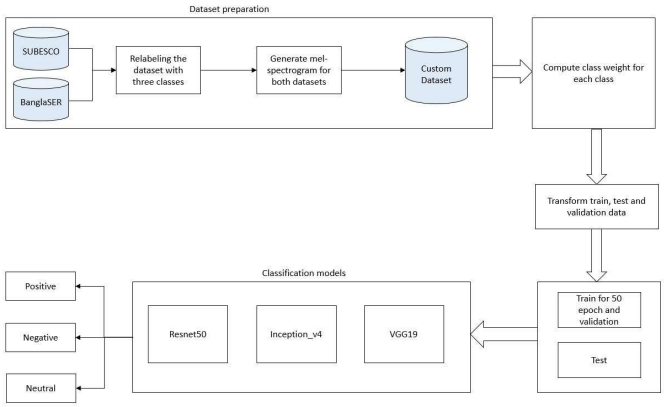



Fig. 3. Proposed Convolutional Neural Network Models Architecture

*1) **VGG-19**:* Pre-trained models can be efficient to use when there is a limited training dataset. VGG-19 proposed in 2014 [18] is a pre-trained CNN architecture reported to achieve high accuracies for image processing large datasets such as ImageNet [19]. It has been trained over a dataset consisting of 1.2 million images with about 100 categories. With the help of its 19 layers, 16 convolution layers, and 3 fully connected layers, it increases the depth of the network and reduces the size of the convolution filter. The convolution layers being 3 * 3 layers helps to determine finer features using the feature maps. A common set of hyperparameters have been used to classify the models. Each convolution layer is stacked by the ReLU activation function. We have also used class weights to combat the imbalanced dataset. VGG-19 helps for faster computation and recognizing the discriminating feature of the mel-spectrogram images classified based on the sentiments.

*2) **Inception v4**:* Inception v4 is an architecture which is published by Google as a paper named "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning" [20] which is an improvement of previously established inception architectures. This also explores the possibility of using residual networks on the inception model named inception ResNet. Inception minimizes the effort of choosing between a convolution layer and a pooling layer and performs both to give a concatenated result. A reduction block is used here to change the height and width of the grid. The stem (initial operations before the inception block) is modified and simplified in inception v4 which gives more computational efficiency.

We have used the transfer learning method of Inception learning but without the pre-trained weights using our transformed spectrogram image dataset from the speech audio. The common hyperparameters mentioned in the experimental setup of table III are used in training the model.

*3) **ResNet-50**:* ResNet-50 [21] is a convolutional neural network that is 50 layers deep and helps to resolve the vanishing gradient problem that occurs in CNN architecture. Its unique skip connection i.e., adding the input to the output and skipping the in-between layers makes a great shift in the output received. This increases the efficiency of the neural network model and skips connection also making it possible to train much deeper networks than previously possible. Moreover, this pre-trained ResNet-50 showed great success with only a 3.57% error rate.

**Long short-term memory (LSTM) and Bidirectional LSTM (Bi-LSTM)**

Hochreiter and Schmidhub first proposed LSTM being a variant of RNN in 1997 [22]. LSTM usually deals with the data stream only in the forward direction whereas Bi-LSTM [23] proposed by Schuster and Paliwal process the audio files in both forward and backward direction. In the case of sequential audio files classical issues such as long-term dependency or short-term memory(vanishing gradient problem) caused in regular RNN network was resolved with these models producing better results. A 3-layer LSTM and Bi-LSTM neural network model has been built using the Keras library with 15 epochs each, a learning rate of 0.001, and a batch size of 32. ReLU and softmax activation function has been used in both LSTM and Bi-LSTM model along with Adam optimizer. The best model during training has been saved for further experiments.

## IV. DATASET

### A. Training Dataset

The right selection of datasets imposes a great impact on the result of models. In this study, we have used the following 2 datasets in the Bengali language to train our models:

**SUBESCO** is a public audio-only emotional speech dataset for Bengali language [8] which consists of 7000 audio speech labeling 7 classes of emotions namely: Happy, Sad, Disgust, Fear, Angry, Neutral, and Surprise. The gender-balanced collection included trained native speakers, and each recording of 10 sentences simulated the seven emotions consisting of an accuracy rate of 71% - 80%The average file length of each audio file is 4s and all the audio files are available in *.wav* format.

TABLE I
TRAIN DATASET CLASS DISTRIBUTION

| Sentiment | Emotions | | No. of Samples | |
|---|---|---|---|---|
| | SUBESCO | BanglaSER | SUBESCO | Bangla SER |
| Positive | Happy | Happy | 2000 | 612 |
| | Surprise | Surprise | 2612 | |
| Negative | Angry | Angry | 4000 | 612 |
| | Sad | Sad | 4612 | |
| | Fear | - | | |
| | Disgust | - | 1000 | 243 |
| Neutral | Neutral | Neutral | 1243 | |

**BanglaSER** [9] consists of 1467 Bangla speech-audio recordings made by an equal number of male and female speakers including 5 different emotions namely angry, happy, neutral, sad, and surprise.

From these two datasets, we have categorized the emotions and labeled each emotion to a specific sentiment based on human understanding. Splitting the emotion label from each audio file, we form our sentiment audio dataset by labeling the sentiment in 3 classes based on the received emotion. The table I demonstrates the classification of the emotions and categorizes them to each sentiment.

### B. Test Dataset

For testing our models we opted for real-life audio files for more accurate evaluation. First, we extracted audio files from the Bengali YouTube videos. From those files, we tailored the speech files in a way so that every single sentence speech audio consists of 3 to 4 seconds to match our training data set. In this way, we created 150 Bangla real audio speech files with noise. Then for annotation, we asked a total of 5 people to annotate the 150 files using our three class labels - Positive, Negative, and Neutral. Then we appended the class labels to the audio files and created the final test dataset.

The dataset consists of both male and female speech audios of different situations. The distribution of the dataset after an annotation is as follows -

TABLE II
TEST DATASET DISTRIBUTION

| Positive | Negative | Neutral |
|---|---|---|
| 43 | 68 | 39 |

## V. EXPERIMENTAL RESULTS AND DISCUSSION

### A. Experimental Setup

For this experiment, we have used Google Colab free GPU for deep learning methods and CPU for ML methods, we have used Tesla T4 GPU for deep learning methods and logged all our results in WandB. We have used python 3 and PyTorch and poutyne for deep learning models. PyTorch Image Models

( timm ) is also used and for the training process, poutyne is used as the wrapper. For ML models, scikit learn, and for data set balance we have used imbalance learn in this experiment.

We divided our dataset into an 80:20 ratio as train: validation set. The train set is used for training all of the models. And the validation set is used for evaluating the models and comparing the results. We have chosen the best model based on the validation set results.

For CNN models we have used fixed hyper parameters in all three of our models and they are as follows -

TABLE III
CNN MODEL HYPERPARAMETERS

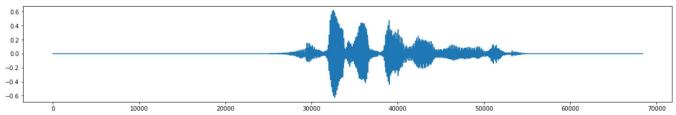
| Parameter | Value |
|---|---|
| imgWidth, imgHeight | 224,224 |
| Epochs | 50 |
| Batch size | 32 |
| Learning Rate | 0.001 |
| Mean | (0.485, 0.456, 0.406) |
| STD | (0.229, 0.224, 0.225) |
| Criterion | CrossEntropyLoss |
| Optimizer | Adam |

### *B. Results*

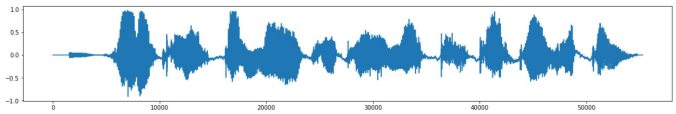
From table IV, we can see the results of our ML and Sequence models on our validation set. The results show that the dataset performs well on ML models in comparison to the other model performances. Among the ML models, RF shows the best results with nearly **90%** accuracy and F1-score with the MFCC feature extraction method, and AdaBoost performs worst with 38% accuracy for the ZCR feature extraction method. If we look at the results of the sequence learning models we can see that the highest accuracy here is only **84%** for the Bi-LSTM model for MFCC feature extraction technique. For all combined features, LSTM and Bi-LSTM also show the prominent result of 79% accuracy.

If we compare the models we can clearly see that the ML models are performing better with the validation set. If we look at table V, CNN models are showing poor results even when it is tested with the validation set. That is why for testing our custom dataset we only considered the ML models and LSTM, Bi-LSTM models.

**Accuracy Score on Custom Test Dataset**



(a) Train Data



(b) Test Data

Fig. 4. Difference between the frequency of Train and Test data

Table VI demonstrates the different ML and LSTM, Bi-LSTM model results on our custom dataset. From the table, we can see that Adaboost among the ML models performs best on the custom test dataset by achieving an accuracy of **45%** with the Zero Crossing Rate feature extraction method whereas the Random Forest model performs worst with 25% accuracy on Chroma Shift extraction method. The weighted average precision of KNN is nearly 43% for combined features and 45% for Bi-LSTM which is satisfactory as the custom dataset was totally unseen and extracted from real-life scenarios. We can see the difference in the frequency between our train and test data in figure 4 where the test data has more frequency, is spread more, and is noisy in comparison to the training set data which is free of noise and closed within a range as it is experimental data.

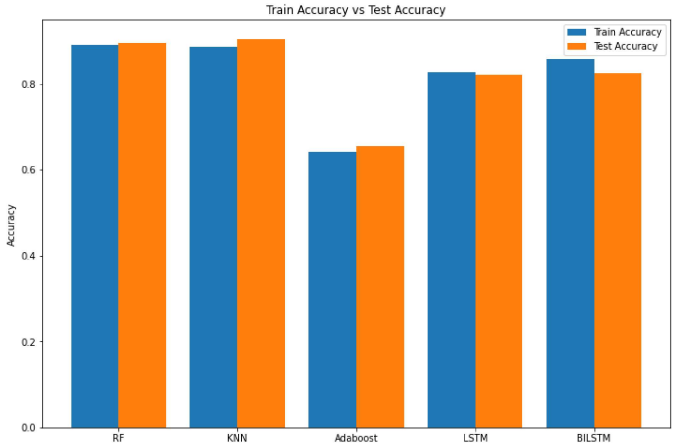
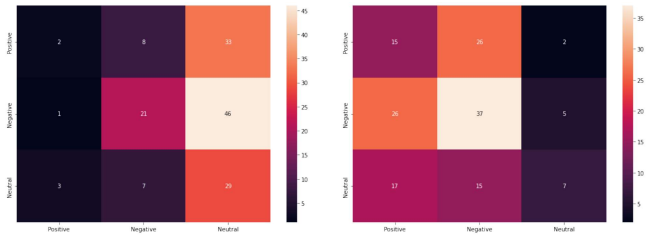
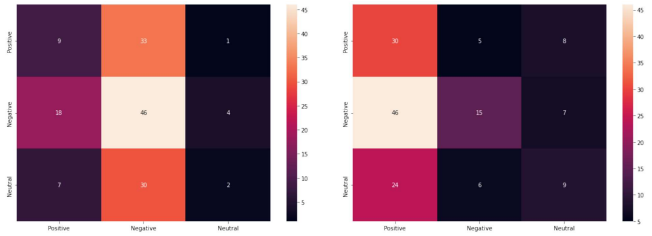



Fig. 5. Train vs Test accuracy of ML models



(a) KNN (b) AdaBoost



(c) Random Forest (d) BiLSTM

Fig. 6. Confusion Matrices for different Models

From the confusion matrices in figure 6 we can see that the best performing model AdaBoost has correctly predicted only 15 positive classes correctly among the 43 data. It correctly

TABLE IV
EVALUATION SCORE ON VALIDATION DATASET

| Features | Evaluation Metric | Machine Learning Models | | | Sequence Models | |
|---|---|---|---|---|---|---|
| | | RF | KNN | AdaBoost | LSTM | Bi-LSTM |
| MFCC | Accuracy | **90%** | 92% | 64% | 81% | 84% |
| | Precison | 90% | **93%** | 62% | 81% | 84% |
| | F1-Score | **90%** | 92% | 62% | 81% | 84% |
| ZCR | Accuracy | 44% | 46% | 38% | 32% | 34% |
| | Precison | 44% | 46% | 38% | 11% | 11% |
| | F1-Score | 44% | 46% | 38% | 16% | 17% |
| Chroma Shift | Accuracy | 76% | 73% | 47% | 32% | 32% |
| | Precison | 76% | 73% | 47% | 10% | 10% |
| | F1-Score | 76% | 71% | 46% | 16% | 16% |
| RMS | Accuracy | 48% | 51% | 47% | 34% | 32% |
| | Precison | 48% | 51% | 46% | 11% | 11% |
| | F1-Score | 48% | 51% | 45% | 17% | 16% |
| Combined Features | Accuracy | **90%** | 90% | 66% | 80% | 80% |
| | Precison | 90% | 91% | 64% | 79% | 81% |
| | F1-Score | 89% | 90% | 64% | 79% | 79% |

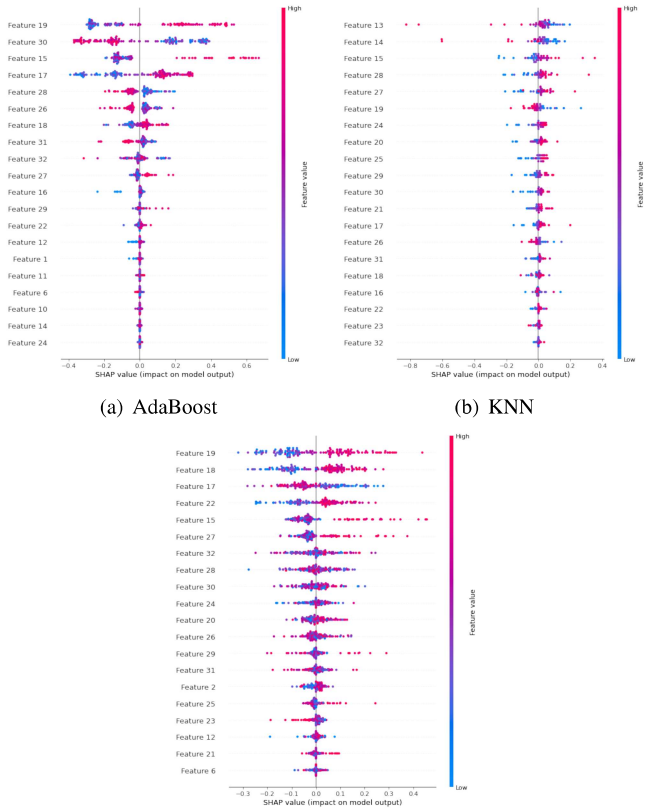
TABLE V
EVALUATION SCORE OF CNN MODELS ON TRAIN DATASET

| CNN Models | | | | |
|---|---|---|---|---|
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| VGG-19 | 58% | 61% | 58% | 53% |
| Inception-v4 | 55% | 45% | 55% | 48% |
| ResNet-50 | 58% | 64% | 58% | 51% |

classified 37 negative data and only 7 neutral data. Most of the neutral data is classified as positive. As in real life, the neutral and positive classes can have overlapping sentiments the model is showing overlapping results during the classification too. Bi-LSTM is mostly classifying the data into the positive class.

### C. SHAP Interpretation of the Proposed ML Models

As our best outputs are received from the ML models, from figure 7 we further depict the distribution impact of features on the test model output. The color represents the feature value (red high, blue low). For AdaBoost in figure 7(a) we can see that after combining the results of all feature extraction methods feature no 19, 30, 15, and 17 play a prominent role in detecting positive and negative sentiment. In the case of RF our best model on the test dataset, figure 7(c) reveals that when feature 19 has a high value shown in red, negative sentiments are predicted and vice versa for blue values. Moreover, compared to KNN and AdaBoost, the distribution of feature impact can be observed more in the RF SHAP beeswarm plot.




(a) AdaBoost (b) KNN

(c) Random Forest

Fig. 7. SHAP Impact on ML Models

## VI. CONCLUSION AND FUTURE WORK

### A. Conclusion

Sentiment Analysis of the Bengali language based on speech signals has become potentially a major topic for research in the field of interaction between humans and computers due to its growing usage in different applications. In this paper, an approach to show a comparative study for sentiment analysis of the Bengali language between ML models and other state of art models has been presented. We have various feature extraction methods to draw a better comparison between the models. Among the models, we observed how the ML models showed better evaluation scores. After finding the best score of 90% based on these evaluation scores we further experimented with KNN, RF, AdaBoost, LSTM, and Bi-LSTM models on our custom annotated test dataset receiving an accuracy of 45% using the AdaBoost model. ML classifiers have proved

TABLE VI
EVALUATION SCORE ON CUSTOM TEST DATASET

| Features | Evaluation Metric | Machine Learning Models | | | Sequence Models | |
|---|---|---|---|---|---|---|
| | | RF | KNN | AdaBoost | LSTM | Bi-LSTM |
| MFCC | Accuracy | 42% | 36% | 37% | 37% | 37% |
| | Precision | **50%** | 38% | 40% | 43% | 43% |
| | F1 Score | 38% | 33% | 35% | 36% | 37% |
| ZCR | Accuracy | 34% | 36% | **45%** | 26% | **45%** |
| | Precision | 33% | 37% | 21% | 7% | 21% |
| | F1-Score | 34% | 36% | 28% | 11% | 28% |
| Chroma Shift | Accuracy | 25% | 29% | 29% | 26% | **45%** |
| | Precision | 25% | 46% | 31% | 07% | 21% |
| | F1-Score | 23% | 24% | 19% | 11% | 28% |
| RMS | Accuracy | 41% | 39% | 47% | 29% | 29% |
| | Precision | 41% | 39% | 36% | 08% | 08% |
| | F1-Score | **41%** | 39% | 36% | 13% | 13% |
| Combined Features | Accuracy | 38% | 35% | 39% | 32% | 36% |
| | Precision | 34% | 43% | 42% | 35% | 45% |
| | F1-Score | 33% | 31% | 38% | 28% | 33% |

to work the best for our training and customized test dataset.

### B. Future Work

Since we have prepared only 150 custom annotated test datasets we hope to increase this to at least 500 audio files and annotate accordingly. Although we experimented with multiple feature extraction methods and acquired satisfying results, other feature extraction techniques may be explored such as MFEC, spectral Roll-off, etc. We also plan to compare the transformer based models to produce a concrete comparison conclusion. Although ML models outperformed CNN, LSTM and Bi-LSTM showed promising results. In the future, we plan to build a novel approach of sentiment analysis with the best-performing methods and models of this comparison study which can outperform the state-of-art models and can contribute to Bengali sentiment analysis research works.

## REFERENCES


[1] V. Vyas and V. Uma, "Approaches to sentiment analysis on product reviews," in *Sentiment Analysis and Knowledge Discovery in Contemporary Business*. IGI global, 2019, pp. 15–30.

[2] M. Wöllmer, F. Weninger, T. Knaup, B. Schuller, C. Sun, K. Sagae, and L.-P. Morency, "Youtube movie reviews: Sentiment analysis in an audio-visual context," *IEEE Intelligent Systems*, vol. 28, no. 3, pp. 46–53, 2013.

[3] N. Altrabsheh, M. M. Gaber, and M. Cocea, "Sa-e: sentiment analysis for education," in *International conference on intelligent decision technologies*, vol. 255, 2013, pp. 353–362.

[4] Z. Jiawa, L. Wei, W. Sili, and Y. Heng, "Review of methods and applications of text sentiment analysis," *Data Analysis and Knowledge Discovery*, vol. 5, no. 6, pp. 1–13, 2021.

[5] S. Maghilnan and M. R. Kumar, "Sentiment analysis on speaker specific speech data," in *2017 international conference on intelligent computing and control (I2C2)*. IEEE, 2017, pp. 1–5.

[6] E. Haddi, X. Liu, and Y. Shi, "The role of text pre-processing in sentiment analysis," *Procedia computer science*, vol. 17, pp. 26–32, 2013.

[7] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE transactions on acoustics, speech, and signal processing*, vol. 28, no. 4, pp. 357–366, 1980.

[8] S. Sultana, M. S. Rahman, M. R. Selim, and M. Z. Iqbal, "Sust bangla emotional speech corpus (subesco): An audio-only emotional speech corpus for bangla," *Plos one*, vol. 16, no. 4, p. e0250173, 2021.

[9] R. K. Das, N. Islam, M. R. Ahmed, S. Islam, S. Shatabda, and A. M. Islam, "Banglaser: A speech emotion recognition dataset for the bangla language," *Data in Brief*, vol. 42, p. 108091, 2022.

[10] M. B. Akçay and K. Oğuz, "Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers," *Speech Communication*, vol. 116, pp. 56–76, 2020.

[11] P. Dhar and S. Guha, "A system to predict emotion from bengali speech," *Int. J. Math. Sci. Comput*, vol. 7, no. 1, pp. 26–35, 2021.

[12] M. M. Rahman, D. R. Dipta, and M. Hasan, "Dynamic time warping assisted svm classifier for bangla speech recognition," *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*, pp. 1–6, 2018.

[13] R. Shaik and S. Venkatramaphanikumar, "Sentiment analysis with word-based urdu speech recognition," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 5, pp. 2511–2531, 2022.

[14] P. Zhao, F. Liu, and X. Zhuang, "Speech sentiment analysis using hierarchical conformer networks," *Applied Sciences*, vol. 12, no. 16, p. 8076, 2022.

[15] Z. Luo, H. Xu, and F. Chen, "Audio sentiment analysis by heterogeneous signal features learned from utterance-based parallel neural network," in *AffCon@ AAAI*, 2019.

[16] U. Ayvaz, H. Gürüler, F. Khan, N. Ahmed, and T. Whangbo, "Automatic speaker recognition using mel-frequency cepstral coefficients through machine learning," 2022.

[17] M. Kattel, A. Nepal, A. Shah, and D. Shrestha, "Chroma feature extraction," in *Conference: chroma feature extraction using fourier transform*, no. 20, 2019.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[20] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.