Multi-level Bangla & English Text Classification using deep Learning and machine Learning

**Introduction:**

A company's e-commerce website is the most popular alternative for businesses because it provides a wide range of online buying and selling possibilities and E-commerce website platforms enable the consumer to make purchases without going to a physical store, in contrast to a regular website, which is often used to check and acquire information [4].Daraz is South Asia’s online shopping marketplace with an active presence in Bangladesh also in Pakistan, Sri Lanka, Myanmar and Nepal. Here people buy and sell products and post their personal experiences, opinions, and reviews. Daraz has more than 2.5 million products and Daraz offers a diverse assortment in categories ranging from consumer electronics to household goods, beauty, fashion, sports equipment and also groceries items. The COVID-19 pandemic, which has led to numerous countries issuing stay-at-home orders among their citizens, has caused a significant increase in online buying. Online shopping has taken over as the primary way for consumers to meet their consumption demands due to the closure of the majority of retail stores and the fear of COVID-19 infections [1].

In Bangladesh, Bangla is a widely spoken language. Apart from Bangladesh, many other countries of the world have Bengali speaking people. In the field of NLP, Bangla natural language processing (BNLP) has gained attention [7]. Text classification or natural language processing has been one of the earliest problems in NLP [6]. On the other hand a large number of E-commerce sites have the comment section that allows expressing the opinion of people in the Bengali language.However, the fact that so little study has been done on bangla sentiment analysis raises serious concerns. Furthermore, English is used in a wide range of industries, such as search engines, social media, and customer service, making it a crucial language for NLP to support. It is also the primary language of many popular online platforms, such as Google, Facebook, and Wikipedia, which generates a large amount of data in English. It is the most common language for scientific publications, therefore it is a focal point for NLP applications in scientific research and knowledge extraction[40,41].

When making judgments on subjects like politics, sports, finances, product reviews, and other topics, sentiment analysis is crucial. Humans are subjective and that's why opinions are important [8]. People have differing perspectives on various products in e-commerce sites. As a

result, their expressions vary depending on the product and its quality, which they like or dislike. We now have a very good chance of deploying Machine and Deep Learning algorithms, as well as Natural Language Processing techniques, to detect cyberbullying for which statement is bullying and which statement is not bullying [9]. We can also use Machine, Deep Learning algorithms and natural language processing techniques in e-commerce sites to detect comments which statement is positive or negative.

Machine Learning algorithms were successfully used to predict and detect in many articles. Machine learning has grown rapidly in recent years in the context of data analysis and computing, allowing applications to function intelligently[10]. To learn from previous experience and detect useful patterns in large, unstructured, and complex datasets, machine learning algorithms employ a variety of statistical, probabilistic, and optimization methods[11].There are several uses for these algorithms, including automatic text categorization[12], breast cancer risk prediction[13], Data augmentation in dermatology image recognition using machine learning[14], Machine Learning for Speech Processing[15], Improving the accuracy of medical diagnosis with causal machine learning[16],Statistical Arbitrage in Cryptocurrency Markets using machine learning[17], classify fake news[18] and many more. Deep learning algorithms are also used to take advantage in many research fields. Like multi-class skin lesion classification considering a binary classification support[19], cellular image analysis [20] and more.

User-generated content (UGC) is a valuable source for understanding online shoppers' emotions.Using text-mining techniques, identifies regarding online retail services in online posts like: product, retailer promotion, delivery, payment, communication, return/refund, and price [2]. With the rapid development of deep learning technology, Convolutional neural networks (CNN) and Long short-term memory (LSTM) have become two of the most popular neural networks [5]. Convolutional neural networks (CNN) were built from the infrared spectra without preprocessing the data using hyperparameter adjustment and saliency map [3]. We can use machine learning and deep learning for text classification. In [21] authors develop Bidirectional long-short term memory(bi-lstm) and CNN model for text classification. They [22] proposed a Coordinated CNN-LSTM-Attention(CCLA) model for text sentiment classification. Wang, Jin, et al.[23] performs Tree-Structured Regional CNN-LSTM Model for Dimensional Sentiment Analysis.

We proposed various approaches for sentiment analysis and automatic text prediction from a bangla and english comment in an e-commerce site. Recently many academic and commercial researchers are currently studying and exploring sentiment analysis and automatic text prediction [5,6,7,8,9]. According to a review of the literature, many academic and commercial researchers have been working on machine learning and deep learning based algorithms on Bangla text. But most of the work has been done on the English text, less work has been done on the Bengali text.

Our main contributions are as follows: Bangla and english comments data collection, cleaning and preprocessing of collected data, preparation of a suitable Bangla and english comments dataset, Data Summary Visualization, A cutting-edge text identification technique based on LSTM, BiLSTM and LSTM-CNN Combine Architecture is presented. In this paper, we build some machine learning models which are Logistic Regression, Decision Tree, Random Forest, Naive Bayes, K-Nearest Neighbor, Support Vector Classifier (SVC), Stochastic Gradient Descent (SGD). We also build LSTM, BiLSTM, Embedding, bidirectional LSTM, Conv1D, GlobalMaxPooling1D, Dense, LSTM-CNN combine Architecture and critically evaluated each classifier's performance using our prepared own dataset by analyzing the values of accuracy, precision, recall, F1-score and loss metrics, Confusion matrix. We also determined Maximum, Minimum and Average Length of a bangla and english comment.

**Related works:**

Related works: Machine learning and deep learning-based models are the two main approaches for senti-ment analysis tasks that are covered in this section. This section provides a quick summary of prior scientific contributions.

In author [24], they evaluated a bi-LSTM based architecture with an attention mechanism and convolution layer for the purpose of classifying text sentiment. In author [2], we saw that authors presented a CNN-based architecture for sentiment analysis. In author [3], they proposed The Long Short Term Memory (LSTM) network combined some complicated sentiment analysis units with its two-layer, bidirectional LSTM architecture. In author [4], we saw that authors suggested that CNN is a viable model for extracting attention from text, and authors carried out a series of experiments. In author [5], completed a significant number of experiments on one-layer convolutional neural networks. In author [6], applied two components: the content polarity and

the overall sentiment content. Content polarity is the difference between the number of positive and negative words, which yields the valence score. This makes it easier to accurately depict the sentiment found in the textual data. In author [7], they recommended using LSTM networks with specific technical analysis indicators to predict stock price in comparison to several baseline models like support vector machines (SVM), random forest (RF), and multi-layer perceptrons. In author [8], recommended an important indication of client happiness is sentiment polarity and It may enable businesses to comprehend their current clients better. In author [9], they proposed the standard CNN and LSTM mixed text categorization approach and to obtain sentence representation, it sent the features of high-level phrase sequences extracted using CNN to the LSTM. In author [10], authors created an Arabic text categorization system that employed CNNs as the classifier and TF-IDF for feature extraction and their method obtained 98.89% accuracy. In author [11], authors conducted sentiment analysis on 4000 manually translated into Bangla,

both positive and negative, movie reviews, and achieved accuracy on the LSTM of 82.42%. In author [12], authors use machine learning to predict feature ratings from the text, and in addition to rating prediction, diagnostics are also taken into consideration. In author [13], they presented a CNN for text classification at the character level that significantly improved classification accuracy. In author [14], presented a unique method that makes use of differential privacy to improve the LSTM model's stock prediction capabilities. In author [15], they offer a comparative analysis for classifying sentiment in Bangla news comments using both traditional SVM and deep learning (LSTM and CNN) algorithms. In author [16], they compare the performance of back-propagation-based neural networks for text classification when compared to other supervised machine learning models.

**Methodology:**

The methodology outlined for this study includes several stages, such as data collection, data preparation, model selection, statistical evaluation, and implementation. These steps are illustrated in Figure 1.

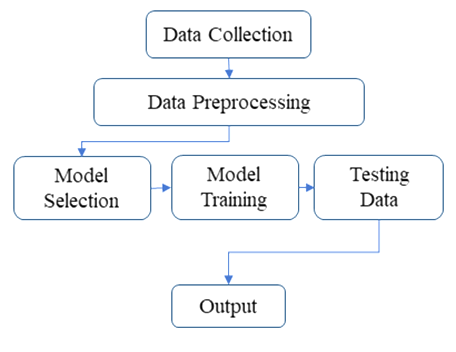


Fig. 1. Methodology at a Glance

**1. Dataset preparation:**

Dataset preparation for natural language processing (NLP) typically involves several steps, including collecting and curating data, categorizing data into different classes, preprocessing and cleaning the data, and splitting the data into training, annotation, balancing data, validation, and test sets. Data preparation includes cleaning and organizing the data for use in NLP tasks, Model selection entails choosing an appropriate algorithm for the task, statistical analysis is utilized to evaluate the model's performance, and the final step is to put the model into action and monitor its performance in the real-world setting. There are two key stages to the dataset preparation process.

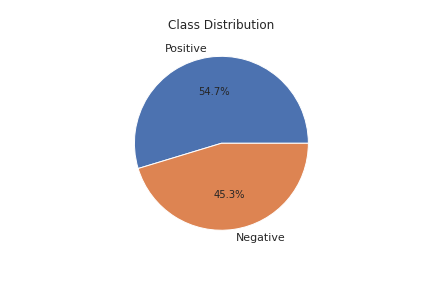
– Data Collection

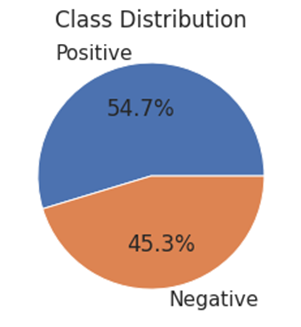
– Preprocessing of Data

**1.1 Data Collection**

Collecting data for a Natural Language Processing (NLP) task is a crucial step. In machine learning, data plays a vital role, and in order to develop models that can be applied more broadly, it is necessary to have a significant amount of data available.

It can be done through web scraping, using publicly available datasets, or by creating your own dataset through manual annotation or crowd-sourcing. We collected our dataset from publicly available datasets. Once the data is collected, we have preprocessed to remove any irrelevant information, correct errors, and format the data in a way that is compatible with the NLP model we plan to use. This includes tokenization, stemming and lemmatization. Finally, we split into training, validation, and test sets, with the test set being reserved for evaluating the performance of the final model. The dataset that we are working with is in CSV format and contains two columns: Sentence and Class. The data has been classified into two categories, namely Negative and Positive. Each row in the dataset has two pieces of information, the first being the Sentence and the second being the Class. The Class attribute indicates whether the sentence is Negative or Positive. We have created a comments-based dataset for natural language processing tasks that includes 1995 examples of English sentences and 1995 examples for Bangla sentences. Our prepared dataset is graphically depicted into two class in fig.1

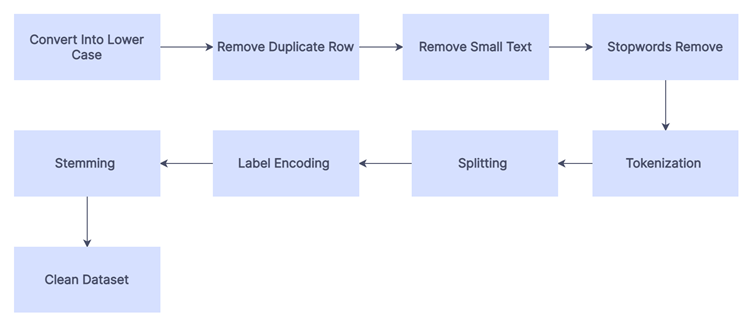






**1.2 Preprocessing of Data:**

Preprocessing is a vital aspect of preparing data for Natural Language Processing tasks. Data preprocessing in natural language processing (NLP) is the process of preparing text data for analysis or modeling. The steps typically include cleaning the text to eliminate irrelevant information and correct errors, dividing the text into words, phrases, or sentences through tokenization, reducing words to their base form through stemming or lemmatization, identifying and removing any data points that are not representative of the data, and normalizing the data to ensure consistency. These steps help to improve the accuracy and efficiency of NLP algorithms. Our contribution for preprocessing of English and Bangla comments data is given below.





**1.2.1 Convert into lower case:** Text normalization in the form of converting all characters to lowercase is a common preprocessing step in sentiment analysis. This process helps simplify the data and make it easier for machine learning models to process and understand the sentiment being conveyed. For sentiment analysis, the datasets contain variations in capitalization, making it challenging for algorithms to identify the sentiment accurately. Converting to lowercase helps eliminate these variations and create a more uniform dataset. Additionally, by reducing the number of unique tokens, the computation required for analysis and model training is decreased, leading to more efficient sentiment analysis.

**1.2.2. Removing Duplicate Rows:** Eliminating duplicate rows is crucial in preprocessing data for NLP tasks as they can cause inaccuracies in the models and increase processing time. We removed duplicate rows in the dataset. The process involves identifying duplicates, choosing a strategy for removing them such as keeping the first instance or the one with the highest confidence score, and using a programming language or tool to implement the strategy and remove the duplicates. By doing so, NLP models' accuracy can be improved, and processing time can be reduced.

**1.2.3. Stopwords Removal:**

Stopwords removal is a frequently used preprocessing technique in NLP tasks. Stopwords are words such as "the," "a," and "and" that do not carry much meaning and can be removed to simplify the dataset and speed up processing. The process involves identifying stopwords, determining a strategy for removal such as removing all or only the most frequently occurring, and using a programming language or software tool to execute the strategy. This helps to make the dataset smaller and processing faster, potentially enhancing the accuracy of NLP models. For the English dataset, we cleaned words like - {“**about**”, “**don’t**”, “**across**”, “**after**”, “**again**”, “**your**”, “**me**”, “**not**”} etc.

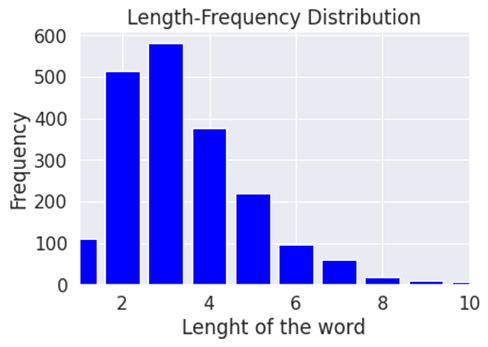
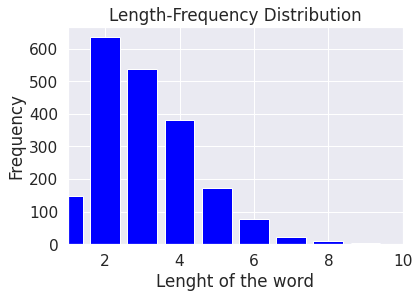
For Bangla dataset, we cleaned word like - {“**এই**”, “**কোন**”, “**আমি**”, “**আপনার**”, “**যা**”, “**যে**”, “**মনে**”, “**করি**”} etc.

**1.2.4 Punctuation, special character and number removal:** Punctuation like; ।**, . ,** ? !, etc. and special character like @, #, $, %, ^, &, \*, etc. and number that is not important for sentiment analysis are removed from the whole Dataset.

**1.2.5. Removing Small Texts:** Filtering small texts, or texts below a certain length, is a crucial step in preprocessing data for NLP tasks. Such texts, such as individual words or brief sentences, might not carry much information and can be removed to improve the quality of the dataset. The process involves determining the minimum length of texts to be kept in the dataset, identifying texts below the threshold, and using a programming language or software tool to remove them. By doing so, the accuracy of NLP models can be enhanced and the quality of the dataset improved. For removing small text, first we determine the length of the sentence (English and Bangla). Then Provide the minimum length of texts to be kept in the dataset. After Cleaning small texts, it Removed 1 Small conversations for English sentences and also removed 1 small conversations for Bangla sentences.

**1.2.6. Stemming:** Stemming is a technique used in NLP to simplify words to their base form. The objective of stemming is to reduce the complexity of the text dataset and make analyzing it easier. For English texts, for example, words like "**recommended**”, "**recommends**", and "**recommendation**" would be transformed to "recommend." For Bangla texts, for example, words like “**করেছিলাম**”, “**করছে**” would be transformed to “**করেছি**”. The process of stemming involves the following steps: extracting the words from the text, utilizing a stemming algorithm and saving the stemmed words in a new data structure for further analysis. Although stemming can help improve the accuracy of NLP models, it can also lead to a loss of information and reduced interpretability.

**1.2.7 Data Set Summary:** A summary of a data set in NLP provides a general overview of the features of the data utilized for training and testing models. Our summary includes details such as the total Number of comments, how the data is distributed across different classes, the number of words, the number of unique words, Most Frequent Words, Average Length of a comment, Maximum Length of a comment, Minimum Length of a comment. Knowing the characteristics of the data set is crucial for selecting the right models and techniques for an NLP task, and for comprehending and evaluating the results of model evaluations. Figure 3.1 and 3.2 provides a clear visual representation of the length-Frequency distribution for English and Bangla texts for the length of the word. Figure 4.1 and 4.2 provides a clear visual representation of the length-Frequency distribution for English and Bangla texts for the length of the character.





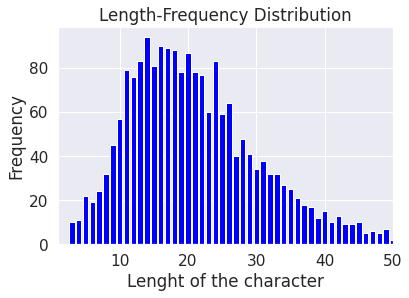
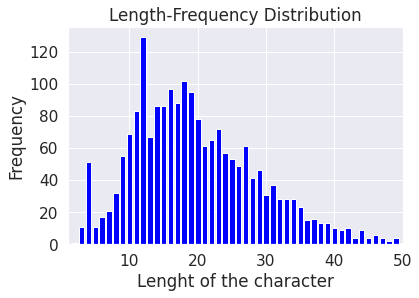


Fig.4.1. Length of character for English text Fig.4.2. Length of character for Bangla

text

**1.2.8 Categorical Encoding:** Categorical encoding is a process in NLP where categorical variables are converted into numerical values that can be used as inputs for machine learning models. Categorical variables are those that have a limited number of possible values, such as words in a vocabulary or items on a menu. There are two types of categorical encoding entitled label encoding and embedding. Label encoding is a way of converting categorical data into numerical format by assigning integer values to each unique category. In our study, label encoding provides a means of transforming categorical data, which cannot be processed directly, into a numerical form. Label encoding assigns each category a unique integer value. The integer values are assigned without implying any ranking or order between the categories. On the other hand, Embedding is a method applied in NLP and machine learning to convert data into dense, low-dimensional vectors. For NLP, words or phrases are transformed into word embeddings, which are high-dimensional vectors that depict their semantic significance and the relationships between words. Word embedding methods rely on the principle that words with similar meanings tend to be used in similar contexts. For example, in our English comment dataset, “**good**” and “**best**” would likely be found in similar contexts and thus have similar meanings. For example, in our Bangla comment dataset, "**ইয়ারফোন**", "**হেডসেট**" and "**হেডফোন**" would likely be found in similar contexts and thus have similar meanings also. The word embeddings of these words would be alike, reflecting the relationships between them. Embedding enhances NLP model performance by encapsulating word meaning and word relationships in a dense, low-dimensional format.

Our English text’s dataset consists of two different classes: Positive and Negative. For Positive comments, there are 1091 total comments, 3331 total words, and 785 unique words. For Negative comments, there are 903 total comments, 2864 total words, and 947 unique words. We have established the values for the number of documents, words, and unique words in the datasets consisting of Positive and Negative data. We have represented these values using a visual representation in figure 5.

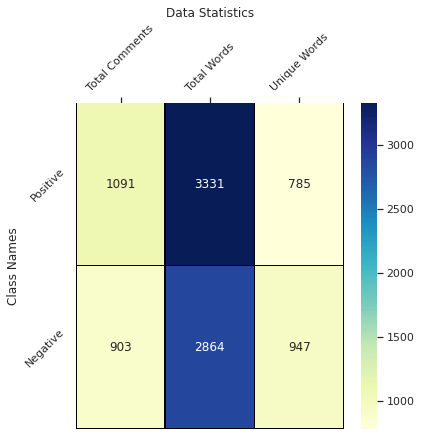


Fig.5. Statistics between class name within sentence for English dataset

Our Bangla text’s dataset consists of two different classes: Positive and Negative. For Positive comments, there are 1090 total comments, 3522 total words, and 1046 unique words. For Negative comments, there are 904 total comments, 3289 total words, and 1224 unique words. We have established the values for the number of documents, words, and unique words in the datasets consisting of Positive and Negative data. We also represented these values using a visual representation in figure 6.

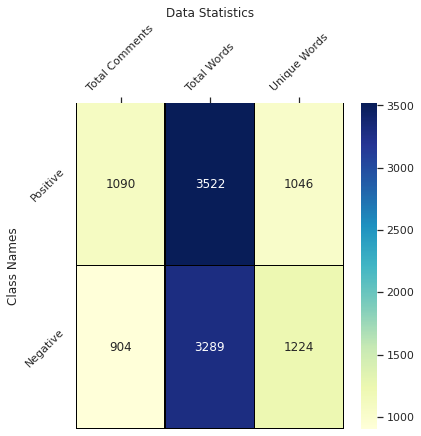


Fig.6. Statistics between class name within sentence for Bangla dataset

**2. Machine Learning Algorithm and Statistical Analysis:**

The dataset was divided using a train-test split method, with the majority of the data (80%) allocated for training the model and the remaining 20% used for testing for both English and Bangla text. The training set consisted of 1567 data points, which were utilized to train various supervised machine learning algorithms, including Support Vector Machine, Multinomial Naive Bayes, K-Nearest Neighbors, Logistic Regression, Decision Tree, Random Forest, and Stochastic Gradient. These trained models were then evaluated using a testing set of 392 data points, with classifier-based algorithms used to determine their accuracy.

**2.1 Feature Extraction:**

In the field of natural language processing, machine learning techniques are employed to accomplish various goals. One such method is known as tokenization, which involves breaking down phrases into their individual word components. These components, both unique and common, are then analyzed for specific characteristics. Another important technique used is Tf-idf, which is a numerical metric that evaluates the importance of a specific term within a text. This approach has been utilized by various reputable publications for multiple languages and has been found to be effective. Our model was inspired by these successful methods and we have found that our learning algorithms are highly accurate when utilizing them.

**2.2 Data Vectorization or Distribution**

The Python scikit-learn library provides a useful tool called CountVectorizer, which can be used to transform a sentence into a vector based on the frequency of each word in the entire text. The size of the n-grams used can be specified using the ngram\_range parameter. For example, a value of 1, 1 would result in unigrams (n-grams made up of a single word), while a value of 1-3 would result in n-grams made up of one to three words.

Unigram: By passing a value of n=1 to the n-grams function, unigrams or 1-grams can be produced, and the word frequency of the words can also be calculated.

Bigram: By passing a value of n=2 to the n-grams function, bigrams or 2-grams can be produced, and the word frequency of the words can also be calculated.

Trigram: By passing a value of n=3 to the n-grams function, trigrams or 3-grams can be produced, and the word frequency of the words can also be calculated.

| Sentence | Unigram | Bigram | Tigrams |
| --- | --- | --- | --- |
| This is the best phone in the budget. | ('the', 2),  ('this', 1),  ('is', 1) | ('this is', 1),  ('is the', 1),  ('the best', 1) | ('this is the', 1),  ('is the best', 1),  ('the best phone', 1) |

Table 1: Example of n-gram distribution for english dataset

| Sentence | Unigram | Bigram | Tigrams |
| --- | --- | --- | --- |
| আমি মনে করি আমি আমার টাকা অপচয় | ('আম', 3),  ('মন', 1),  ('কর', 1) | ('আম মন', 1),  ('মন কর', 1),  ('কর আম', 1) | ('আম মন কর', 1), ('মন কর আম', 1), ('কর আম আম', 1) |

Table 1: Example of n-gram distribution for Bangla dataset

**3. Model and performance:**

**3.1 Proposed Model:**

Using various machine learning techniques, we classified Bengali texts into three groups: simple, complex, and compound. We employed six of the most relevant classification methods, such as Support Vector Machines, Logistic Regression, Decision Trees, Random Forests, K-Nearest Neighbors, and Stochastic Gradient Descent.

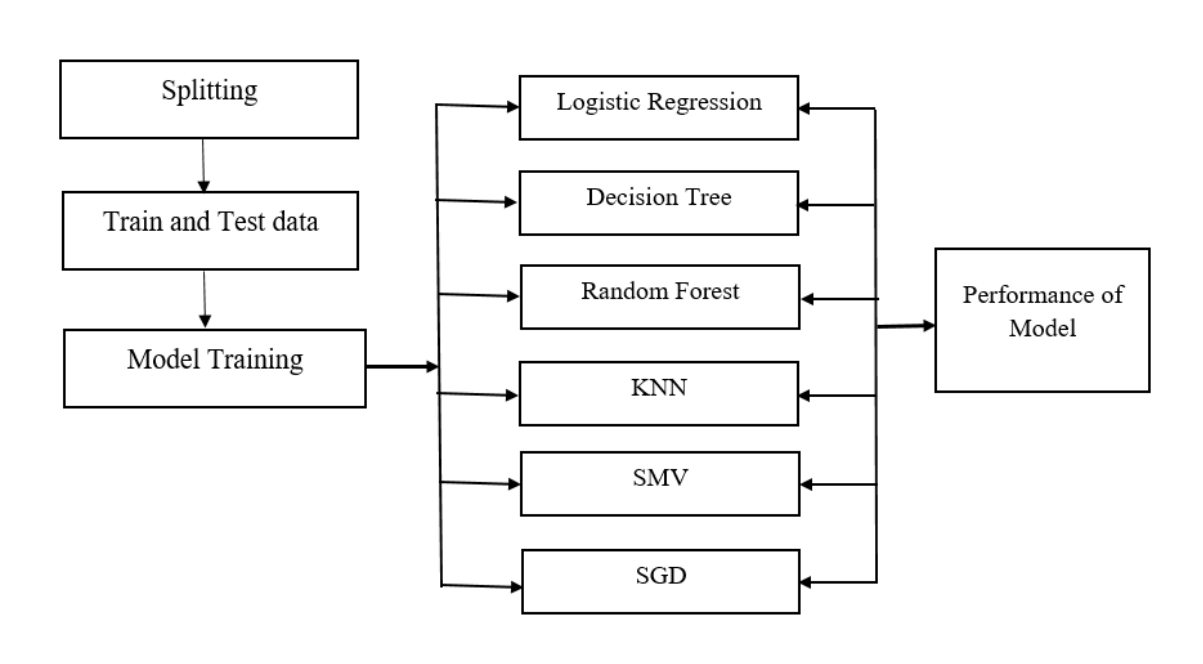


Fig. 7. Working form input

**3.2Model Performance:**

We used several classification techniques to analyze Bengali text data and obtained the following results:

* The Logistic Regression Classifier had an accuracy of 83.16%, which is a supervised method used to link dependent and independent variables.
* The Decision Tree Classifier had a prediction accuracy of 79.34%, which is commonly used for both classification and regression problems.
* The Random Forest Classifier, a system made up of multiple decision trees, was effective in handling high-dimensional data and had a good accuracy of 82.14%.
* The K-Nearest Neighbor (KNN) Classifier, which is used for regression and text processing. It classifies new data by determining its similarity to existing data and assigning it to the most similar category. Our KNN model had an accuracy of 80.10%.
* The Support Vector Machine (SVM) Classifier, which is a powerful algorithm that does not require a large amount of training data to produce accurate results. SVMs classify data by drawing a hyperplane that separates the categories. Our dataset achieved an accuracy of 82.91% with this method.
* The Stochastic Gradient Descent (SGD) Classifier, which is an optimization procedure frequently used in machine learning to identify the model parameters that best match the expected and actual outputs. Our method using SGD achieved an accuracy of 83.67%.

We also provided the results in a table with accuracy, precision, recall, and f1-score.

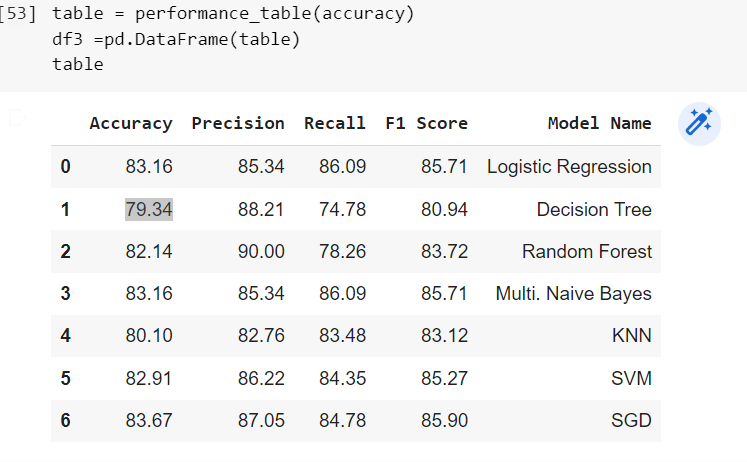


Table 2: accuracy, precision, recall, and f1-score with model names

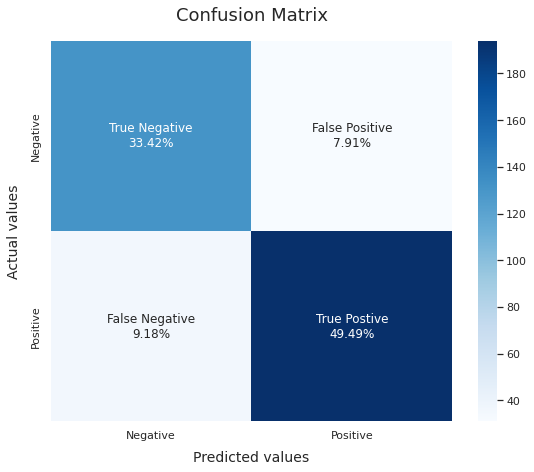


Fig.11. Confusion Matrix on prediction

**3.3 DNN-Based Models**

DNN stands for Deep Neural Network, it's a type of artificial neural network that is composed of multiple layers, called hidden layers, in addition to input and output layers. These hidden layers allow the network to learn and represent complex patterns and relationships in the data, making DNNs particularly useful for tasks such as image and speech recognition, natural language processing, and other tasks that involve high-dimensional data.

**3.3.1 CNN (Convolutional Neural Network):** This is a type of DNN that is particularly well-suited for image processing and other tasks that involve grid-like data. In a CNN, the input data is passed through multiple convolutional and pooling layers, which are designed to detect and extract features from the data. These features are then passed on to one or more fully connected layers for classification or other tasks.

**3.3.2 LSTM (Long Short-Term Memory):** This is a type of DNN that is particularly well-suited for sequential data, such as time series data, speech, and text. LSTM networks are composed of LSTM cells, which are designed to remember information for a long period of time and to selectively forget irrelevant information. This allows LSTM networks to learn patterns in the data that span multiple time steps, making them useful for tasks such as language modeling and speech recognition.

Both CNNs and LSTMs are often used in combination with other types of DNNs to improve the performance on various tasks.

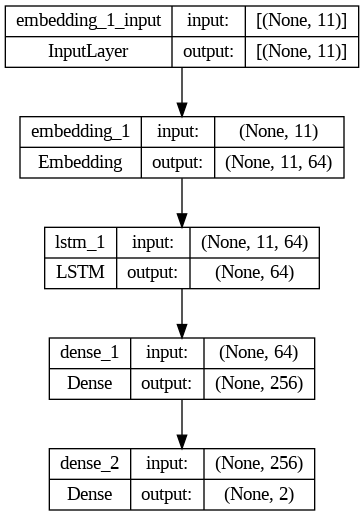
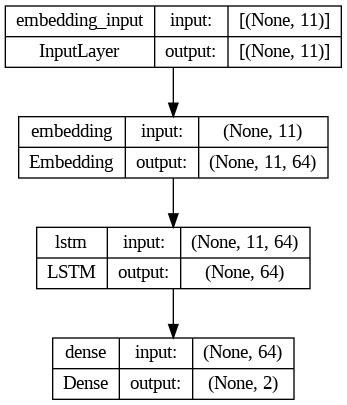
**3.4 Loss Function:** In order to determine how well a model is performing during training, loss functions calculate a value that the model should strive to minimize. In this research, as there are only two possible label classes (0 and 1), the binary cross-entropy loss function was utilized. This type of loss function calculates the difference between predicted labels and true labels. For each example, the loss is determined by computing an average of a single floating-point value per prediction. The binary cross-entropy function calculates the loss of an example by computing the average of this value.

L = -(1/N) \* Σ(y\_i \* log(y\_hat\_i) + (1 - y\_i) \* log(1 - y\_hat\_i))

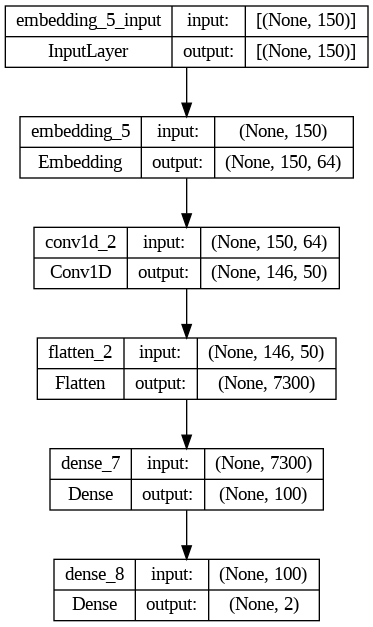
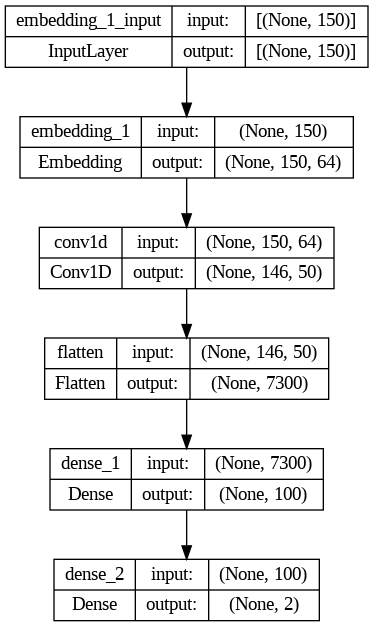
Where N is the number of samples, y\_i is the true label, y\_hat\_i is the predicted label, and log is the natural logarithm.

**3.5 Optimizer:** In this research, the Adam optimization algorithm was used as the optimization method. This algorithm is a variant of stochastic gradient descent that uses adaptive estimates of first and second moments of the gradients to adjust the learning rate. It is computationally efficient, requires limited memory, and is robust to the scaling of gradients. This makes it well-suited for large-scale data and complex models. The Adam optimization algorithm is a widely used optimization technique in machine learning and deep learning tasks.

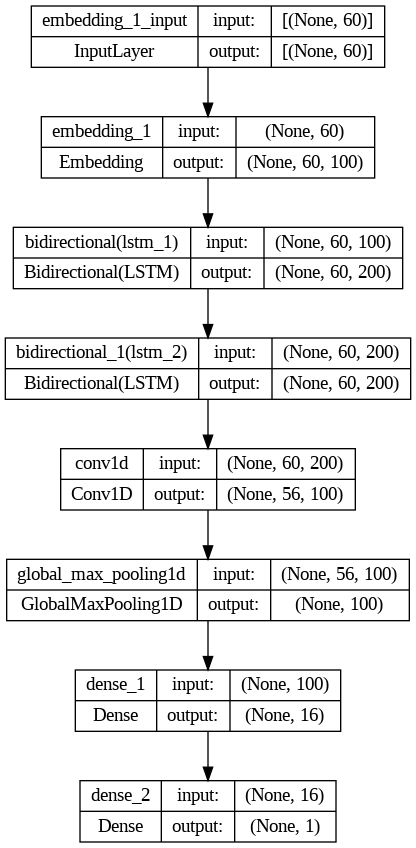
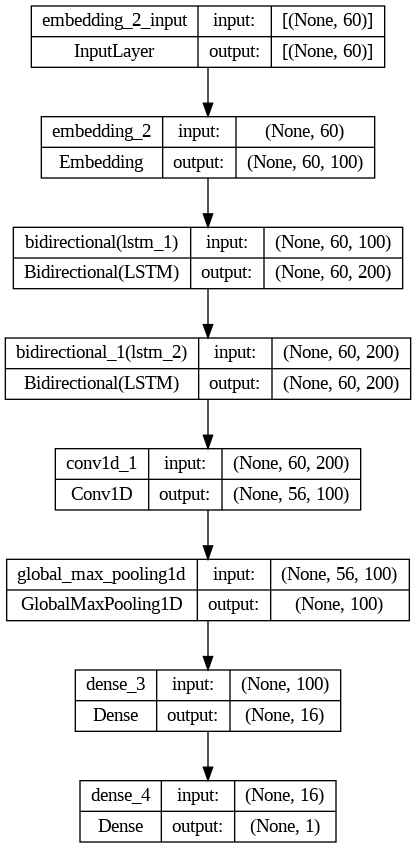
**3.6 Model Configurations:** In this study, five different deep neural network models were utilized, each based on Long Short-Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), and 1D Convolution (Conv1D) architecture.











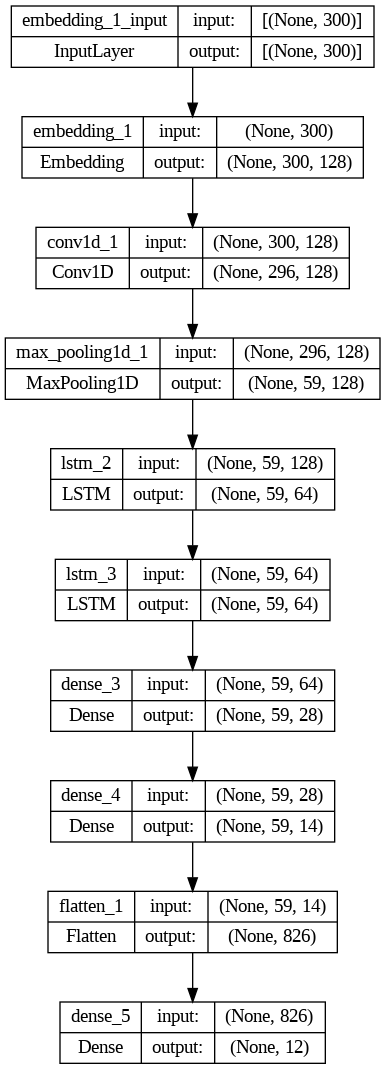
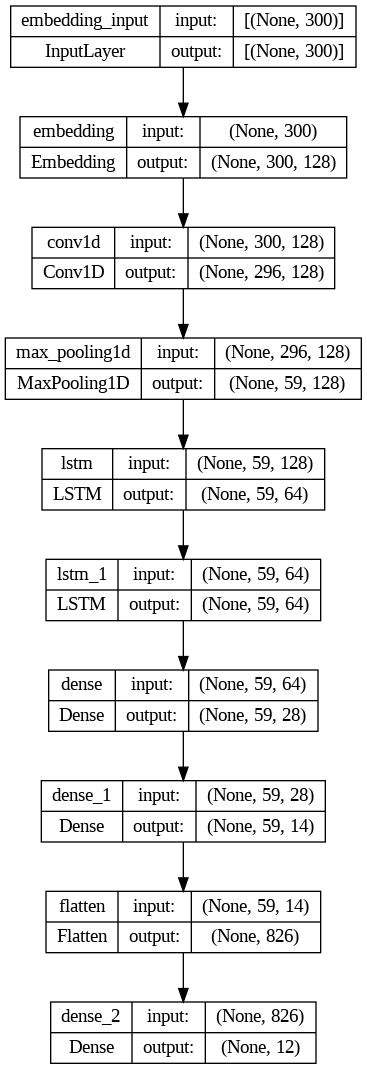




Fig.12. (a–d): Layer visualization of four different DNN models

| Model Name | Embedding  Layer | Conv1D Layer | MaxPooling1D Layer | LSTM Layer | Bi-LSTM Layer | Fully Connected Layer | Dropout Layer | Classification Layer |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| LSTM Based Model | 64 | N/A | N/A | Layer: 1 Unit: 64 | N/A | Layer: 1 Unit: 256 | Layer: 1 (40%) | Softmax |
| Bi-LSTM Based Model | 60 | Layer: 1 Unit: 100 | Layer: 2  Pooling Size: 2 | N/A | Layer: 1 Unit: 100  Layer: 2 Unit: 200 | Layer: 3 Unit: 16 | Layer: 1 (40%) | Softmax |
| Conv1D Based Model | 64 | Layer: 1 Unit: 50 | N/A | N/A | N/A | Layer: 1 Unit: 100 | Layer: 1 (40%) | Softmax |
| Conv1D & Stacked Bi-LSTM Based Model | 128 | Layer: 1 Unit: 50 | Layer: 1 Pooling Size: 2 | Layer: 1 Unit: 64(L1)  64(L2) |  | Layer: 1 Unit: 28 | Layer: 1 (20%) | Softmax |

Table 2. Experimental setting of different DNN models.

**4.4. Experimental Setup**

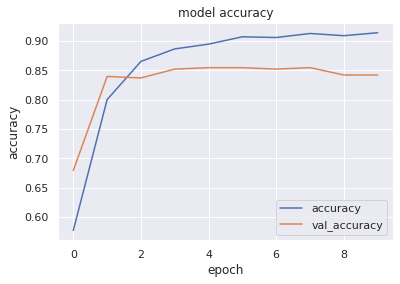
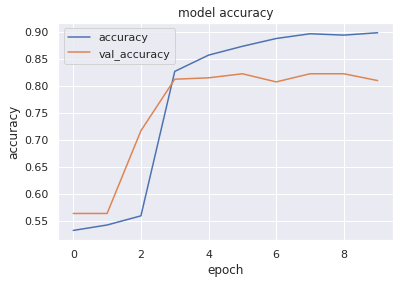
To study the adaptability and improvement of various deep learning algorithms, we conducted experiments with four different classifiers. These classifiers were trained and tested on a dataset of 1995 data points, with a train-test split of 80-20, respectively. The experiments were performed using Python 3.8, with the Keras library and TensorFlow 2.4.1 backend for training the classifiers. The Pandas library was used for handling and preprocessing the data, Matplotlib for visualizing the data, and scikit-learn for calculating performance metrics. To evaluate the performance of our approach, we conducted four different experiments, as outlined in Table 3, with no overlap between the training and test datasets.

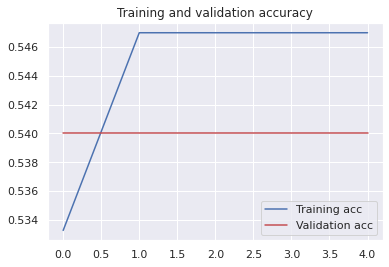
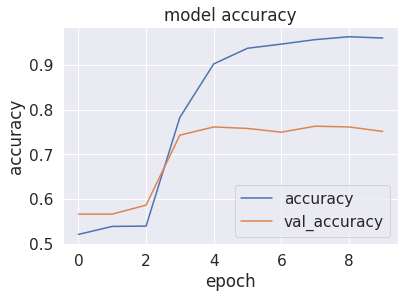
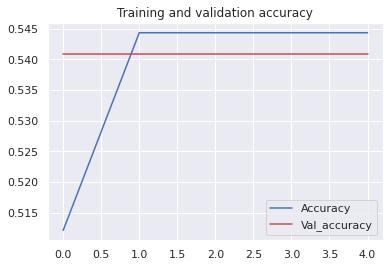
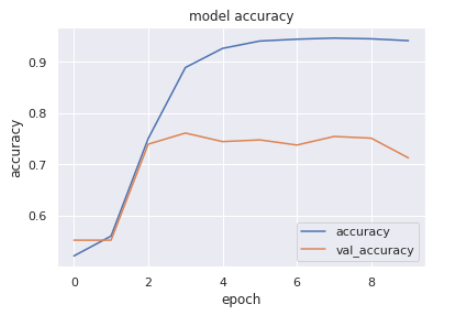
| **Data Type** | **Data Point** |
| --- | --- |
| Training |  |
| Test |  |

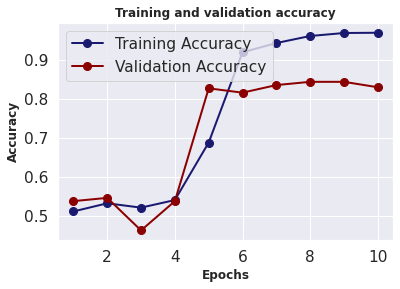
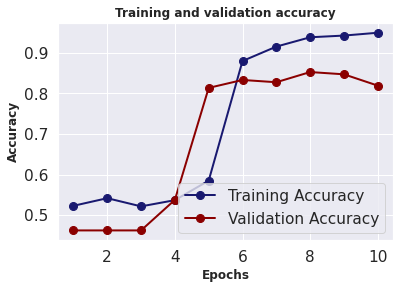
Table 3. Splitting of dataset for training and testing.

**4.5. Prediction and Performance Analysis**

The experiment was carried out to Classify the Sentiment from the Dataset using a training dataset to construct experimental models. Essential parameters were tuned to improve accuracy and reliability. The models were tested using a test dataset to evaluate their performance in terms of various metrics such as accuracy, precision, recall, etc. The results, both in tables and graphs, showed that Conv1D-based DNN had the best performance with an accuracy of 94.01% for the training dataset and 77.21% for the test dataset. The results also indicated that Conv1D and Conv1D combined with LSTM-based DNNs were more effective than simple LSTM-based DNNs, which had the worst performance among the 4 variations of CNN and LSTM-based models.











| **Model Name** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| --- | --- | --- | --- | --- |
| LSTM-Based Model |  |  |  |  |
| Bi-LSTM-Based Model |  |  |  |  |
| Conv1D-Based Model |  |  |  |  |
| Conv1D and LSTM-Based Model |  |  |  |  |

Table 4. The performance of sentiment detection from English dataset

**Experimental Results and Discussion**

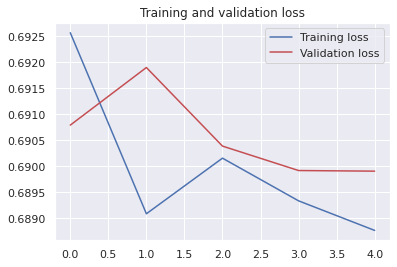
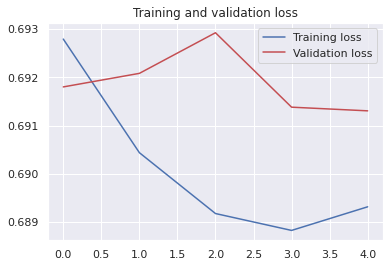
In this study, 4 deep learning classifiers (LSTM, Bi-LSTM, Conv1D, and Conv1D and LSTM combine) were tested using a custom dataset for extracting information. The performance of the models was evaluated using 4 metrics: accuracy, precision, recall and F1-score. The results showed that the Conv1D-based model performed the best, achieving the highest accuracy (92.16%) and F1 score (0.920). The LSTM-based model had the lowest performance with accuracy of 71.57% and an F1 score of 0.773. The results suggest that the Conv1D-based classifier is robust and provides the best output among all tested models. The code and trained models can be found at a public GitHub repository. The performance metrics values of the four proposed systems are displayed clearly in Tables 4 and 5. The comparison of these metrics can also be seen in these tables. Table 6 provides a summary of the experimental results. The results in Table 4 demonstrate that the ConvNet-based model surpasses all others in terms of accuracy, precision, recall and F1-score performance metrics. Table 5 highlights that the Conv1D-based model has a high prediction accuracy for the training and test datasets.

| **Classification Model Name** | **Training** | **Test** |
| --- | --- | --- |
| LSTM-Based Model |  |  |
| Bi-LSTM-Based Model |  |  |
| Conv1D-Based Model |  |  |
| Conv1D and LSTM-Based Model |  |  |

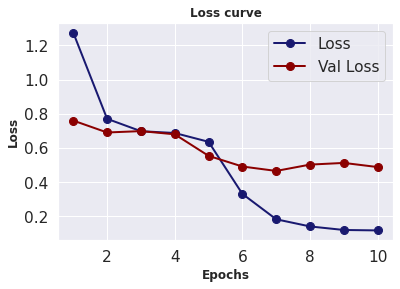
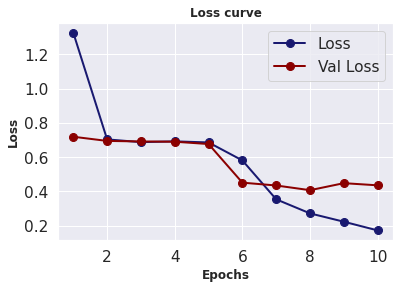
Table 5. Accuracy of five different classifiers for training, validation, and test dataset

| Best Achieved Accuracy | Best Performing Classifier | Second Best Performing Classifier | Least Accuracy | Least Performing Classifier | Best F1 Score |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

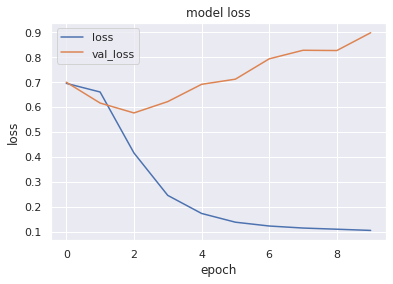
Table 6. Comparison among the experimental results



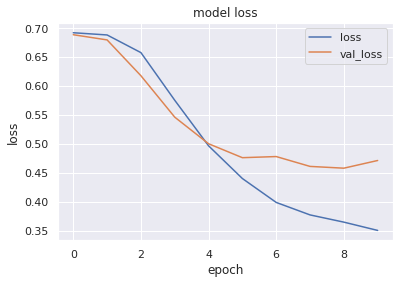


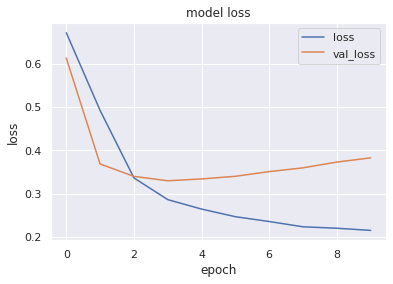














**References:**

[1] Balakrishnan, V., Shi, Z., Law, C.L. *et al.* A deep learning approach in predicting products’ sentiment ratings: a comparative analysis. *J Supercomput* 78, 7206–7226 (2022). <https://doi.org/10.1007/s11227-021-04169-6>.

[2] Jia-Jhou Wu, Sue-Ting Chang, Exploring customer sentiment regarding online retail services: A topic-based approach, Journal of Retailing and Consumer Services, Volume 55,

2020, 102145, ISSN 0969-6989, <https://doi.org/10.1016/j.jretconser.2020.102145>.

[3] Determination of the lactose content in low-lactose milk using Fouriertransform infrared spectroscopy (FTIR) and convolutional neural network, HELIYON (2023), [doi: https:// doi.org/10.1016/j.heliyon.2023.e12898](https://www.cell.com/heliyon/pdf/S2405-8440(23)00105-6.pdf).

[4] Sarvjeet Kaur Chatrath, G.S. Batra, Yogesh Chaba, “Handling consumer vulnerability in e-commerce product images using machine learning”, HELIYON [VOLUME 8, ISSUE 9](https://www.cell.com/heliyon/issue?pii=S2405-8440(21)X0010-2), E10743, SEPTEMBER 01, 2022, <https://doi.org/10.1016/j.heliyon.2022.e10743>.

[5] Yuandong Luan, Shaofu Lin, “Research on Text Classification Based on CNN and LSTM”, 978-1-7281-1223-7/19/$31.00 ©2019 IEEE.

[6] Alam, Tanvirul, Akib Khan, and Firoj Alam. "Bangla text classification using transformers." *arXiv preprint arXiv:2011.04446* (2020).

[7] Bhowmik, Nitish Ranjan, Mohammad Arifuzzaman, and M. Rubaiyat Hossain Mondal. "Sentiment analysis on Bangla text using extended lexicon dictionary and deep learning algorithms." Array 13 (2022): 100123.

[8] Moqsadur Rahman, Summit Haque, Zillur Rahman Saurav, “Identifying and Categorizing Opinions Expressed in Bangla Sentences using Deep Learning Technique”, International Journal of Computer Applications (0975 – 8887), Volume 176 – No. 17, April 2020.

[9] Md. Tofael Ahmed, Maqsudur Rahman, Shafayet Nur, Azm Islam and Dipankar Das, “Deployment of Machine Learning and Deep Learning Algorithms in Detecting Cyberbullying

in Bangla and Romanized Bangla text: A Comparative Study”, 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT) | 978-1-7281-5791-7/20/$31.00 ©2021 IEEE, DOI: 10.1109/ICAECT49130.2021.9392608

[10] Sarker IH. Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. SN Comput Sci. 2021.

[11] Mitchell, Tom M., and Tom M. Mitchell. *Machine learning*. Vol. 1. No. 9. New York: McGraw-hill, 1997.

[12] Sebastiani, Fabrizio. "Machine learning in automated text categorization." *ACM computing surveys (CSUR)* 34.1 (2002): 1-47.

[13] Ming, C., Viassolo, V., Probst-Hensch, N. *et al.* Machine learning techniques for personalized breast cancer risk prediction: comparison with the BCRAT and BOADICEA models. *Breast Cancer Res* 21, 75 (2019). <https://doi.org/10.1186/s13058-019-1158-4>.

[14] Aggarwal, 1st Lt Pushkar. "Data augmentation in dermatology image recognition using machine learning." *Skin Research and Technology* 25.6 (2019): 815-820.

[15] Jung, Young Hoon, et al. "Flexible piezoelectric acoustic sensors and machine learning for speech processing." Advanced Materials 32.35 (2020): 1904020.

[16] Richens, Jonathan G., Ciarán M. Lee, and Saurabh Johri. "Improving the accuracy of medical diagnosis with causal machine learning." *Nature communications* 11.1 (2020): 1-9.

[17] Fischer, Thomas Günter, Christopher Krauss, and Alexander Deinert. "Statistical arbitrage in cryptocurrency markets." *Journal of Risk and Financial Management* 12.1 (2019): 31.

[18] Hakak S, Alazab M, Khan S, Gadekallu TR, Maddikunta PK, Khan WZ. An ensemble machine learning approach through effective feature extraction to classify fake news. Future Generation Computer Systems. 2021 Apr 1;117:47-58.

[19] Balazs Harangi, Agnes Baran, Andras Hajdu,

“Assisted deep learning framework for multi-class skin lesion classification considering a binary classification support”,Biomedical Signal Processing and Control, Volume 62, 2020, 102041,ISSN 1746-8094, <https://doi.org/10.1016/j.bspc.2020.102041.>

[20] Moen, Erick, Dylan Bannon, Takamasa Kudo, William Graf, Markus Covert, and David Van Valen. "Deep learning for cellular image analysis." Nature methods 16, no. 12 (2019): 1233-1246.

[21] Jang, Beakcheol, Myeonghwi Kim, Gaspard Harerimana, Sang-ug Kang, and Jong Wook Kim. "Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism." *Applied Sciences* 10, no. 17 (2020): 5841.

[22] Zhang, Y.; Zheng, J.; Jiang, Y.; Huang, G.; Chen, R. A Text Sentiment Classification Modeling Method Based on Coordinated CNN-LSTM-Attention Model. Chin. J. Electron. 2019, 28, 120–126.

[23] Wang, Jin, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. "Tree-structured regional CNN-LSTM model for dimensional sentiment analysis." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2019): 581-591.

[24] Liu, G., Guo, J., 2019. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. Neurocomputing 337, 325–338.

[25] Dos Santos, C., Gatti, M., 2014. August). Deep convolutional neural networks for sentiment analysis of short texts. In: In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp.69–78.

[26] K. S. Tai, R. Socher, and C. D. Manning, “Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks,” ACL-IJCNLP 2015 - 53rd Annu. Meet. Assoc. Comput. Linguist. 7th Int. Jt. Conf. Nat. Lang. Process. Asian Fed. Nat. Lang. Process. Proc. Conf., vol. 1, pp. 1556–1566, Feb. 2015, doi: 10.48550/arxiv.1503.00075.

[27] J. Du, L. Gui, R. Xu, and Y. He, "A Convolutional Attention Model for Text Classification," in National CCF Conference on Natural Language Processing and Chinese Computing, 2017: Springer, pp. 183-195.

[28] Y. Kim, "Convolutional neural networks for sentence classification," arXiv preprint arXiv:1408.5882, 2014.

[29] Chatterjee, S. (2020). Drivers of helpfulness of online hotel reviews: A sentiment and emotion mining approach. International Journal of Hospitality Management, 85, Article

102356. https://doi.org/10.1016/j.ijhm.2019.102356.

[30] Nelson, D., Pereira, A., Oliveira, R.: Stock market’s price movement prediction with LSTM neural networks. In: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 1419–1426(2017). https://doi.org/10.1109/IJCNN.2017.7966019

[31] E. Park, “Motivations for customer revisit behavior in online review comments: Analyzing the role of user experience using big data approaches,” J. Retail. Consum. Serv., vol. 51, no. May, pp. 14–18, 2019, doi: 10.1016/j.jretconser.2019.05.019.

[32] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, “A C-LSTM Neural Network for Text Classification,” 2015, [Online]. Available: http://arxiv.org/abs/1511.08630

[33] M. Alhawarat and A. O. Aseeri, “A Superior Arabic Text Categorization Deep Model ( SATCDM ),” vol. 8, pp. 24653–24661, 2020.

[34] R. R. Chowdhury, M. S. Hossain, S. Hossain, and K. Andersson, “Analyzing Sentiment of Movie Reviews in Bangla by Applying Machine Learning Techniques,” pp. 27–28, 2019.

[35] Chakraborty, I., Kim, M., & Sudhir, K. (2021). Attribute sentiment scoring with online text reviews: Accounting for language structure and attribute self-selection. Cowles Foundation Discussion Paper, No. 2176R2.

[36] X. Zhang, J. Zhao, and Y. LeCun, "Character-level

convolutional networks for text classification," in Advances in

neural information processing systems, 2015, pp. 649-657.

[37] Li, X., Li, Y., Yang, H., Yang, L., Liu, X.: DP-LSTM: differential privacy-inspired LSTM for stock prediction using financial news.arXiv:1912.10806 (2019)

https://doi.org/10.1109/ACCESS.2019.2895252.

[38] S. Shovon and S. Haque, “Data Set For Sentiment Analysis On Bengali News Comments And Its Baseline Evaluation,” pp. 27–28, 2019.

[39] M. Rahman, S. Haque, and Z. Rahman, “Identifying and Categorizing Opinions Expressed in Bangla Sentences using Deep Learning Technique,” Int. J. Comput. Appl., vol. 176, no. 17, pp. 13–17, 2020, doi: 10.5120/ijca2020920119.

[40] Dreisbach, Caitlin, Theresa A. Koleck, Philip E. Bourne, and Suzanne Bakken. "A systematic review of natural language processing and text mining of symptoms from electronic patient-authored text data." International journal of medical informatics 125 (2019): 37-46.

[41] Ferreira‐Mello, Rafael, Máverick André, Anderson Pinheiro, Evandro Costa, and Cristobal Romero. "Text mining in education." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9, no. 6 (2019): e1332.