

**BENGALI FUNCTIONAL SENTENCE CLASSIFICATION THROUGH
MACHINE LEARNING APPROACH**

BY

**ANTARA BISWAS
ID: 181-15-10644**

**MUSFIQUR RAHMAN
ID: 181-15-10872**

**ZAHURA JEBIN ORIN
ID: 181-15-11127**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By

Md. Zahid Hasan
Associate Professor
Department of CSE
Daffodil International University

Co-Supervised By

Mr. Md. Azizul Hakim
Sr. Lecturer
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY


DHAKA, BANGLADESH

DECEMBER 2021

APPROVAL

This Project titled “**Bengali Functional Sentence Classification through Machine Learning Approach**”, submitted by Antara Biswas, ID No:181-15-10644, Musfiqur Rahman, ID No:181-15-10872 and Zahura Jebin Orin, ID No:181-15-11127 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 4 January, 2022.

BOARD OF EXAMINERS



Dr. Touhid Bhuiyan
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Dr. Fizar Ahmed
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Nusrat Jahan
Senior Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Mohammad Shorif Uddin
Professor

Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

©Daffodil International University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Md. Zahid Hasan, Associate Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Md. Zahid Hasan
Associate Professor
Department of CSE
Daffodil International University

Co-Supervised by:

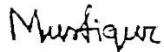


Mr. Md. Azizul Hakim
Sr. Lecturer
Department of CSE
Daffodil International University


Submitted by:



Antara Biswas
ID: 181-15-10644
Department of CSE
Daffodil International University



Musfiqur Rahman
ID: 181-15-10872
Department of CSE
Daffodil International University



Zahura Jebin Orin
ID: 181-15-11127
Department of CSE
Daffodil International University
©Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to Almighty God for His divine blessing making us possible to complete the final year project successfully.

We are grateful and wish our profound indebtedness to **Md. Zahid Hasan, Associate Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Machine Learning*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor Dr. Touhid Bhuiyan, Professor, and Head, Department of CSE, for his kind help to finish our project and also to other faculty members and the staff of the CSE department of Daffodil International University.

We would like to thank our entire coursemate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In the early days, very few studies were effectuated in Bengali Languages as well as its functional sentence. However, studies on Bengali have grown exponentially due to structural diversity. Inspired by these studies, the classification of the Bengali functional sentences was completed with machine learning methods for classifying sentences. The study looked at three different forms of Bengali functional sentences: assertive, interrogative, and exclamatory. Thus, The study's major goal is to categorize the sentence and compare the rate of accuracy to determine the optimal model. The dataset has been properly collected, classified, and processed to avoid conflicts. Some conventional machine learning (ML) classifiers such as Decision Tree (DT), Naive Bayes (NB), Support Vector Machine (SVM), Random Forest (RF), K-Nearest Neighbor (KNN), and Extreme Gradient Boosting(XGB) have been applied to compare classification accuracy rates. Parameters such as precision, recall, F1-score, support, and confusion matrix were calculated for comparison. The comparison proved that the performance of RF, SVC, and XGB classifiers is better than Naive Bayes and Decision Tree classifiers. A notable enigma is that the RF algorithm implemented the highest attainment value with 75.38% accuracy which is the ordinary performance of such datasets.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgments	iii
Abstract	iv
 CHAPTER	
CHAPTER 1: INTRODUCTION	1 - 4
1.1 Introduction	1
1.2 Motivation	2
1.3 Research Objectives	3
1.4 Report Layout	4
CHAPTER 2: RELATED WORKS	5 - 7
2.1 Related Works	5
2.2 Scops of the Problem	6
2.3 Challenges	7
CHAPTER 3: RESEARCH METHODOLOGY	8 - 21
3.1 Introduction	8
3.2 Class Selection	8
3.3 Data Collection Process	9

3.4 Data Preprocessing	11-14
3.4.1 Insertion of Dataset	12
3.4.2 Dataset Diagnosing	12
3.4.3 Dataset Cleaning	13
3.4.4 Future Extraction	14
3.5 Machine Learning Model Selection	14-18
3.5.1 Naive Bayes (NB)	14
3.5.2 Random Forest (RF)	15
3.5.3 Decision Tree (DT)	16
3.5.4 Support Vector Machines (SVM)	17
3.5.5 K-nearest Neighbor (KNN)	18
3.5.6 Extreme Gradient Boosting (XGB)	18
3.6 Cross-Validation	19-20
3.7 Performance Parameters	20-21
3.7.1 Confusion Matrix	20
3.7.2 Precision	20
3.7.3 Recall	21
3.7.4 F1-Score	21
3.7.5 Accuracy	21
CHAPTER 4: RESULT ANALYSIS	22 - 28
CHAPTER 5: CONCLUSION AND FUTURE WORK	29
5.1 Conclusion	29

5.2 Future Works	29
REFERENCES	30 - 32
APPENDIX	33 - 40
PLAGIARISM REPORT	41 - 46

LIST OF FIGURES

FIGURES	PAGE NO
Figure 1.2: Fields of Artificial Intelligence	2
Figure 3.1: The Flow Chart of the Proposed Methodology	8
Figure 3.3.1: Diagram of Data Collection	9
Figure 3.3.2: Pie Chart View of Collected Data	10
Figure 3.4: Diagram of Data Preprocessing	12
Figure 3.6: K-Fold Cross Validation where $k=10$ and the data set are randomly split into 10 detach subsets.	19
Figure 3.7.1: Confusion Matrix	20
Figure 4.1: Comparison between test and cross-validation accuracy of different machine learning techniques	22
Figure 4.2: Comparison among precision, recall, and f1-score for class 0 (Interrogative Sentence)	27
Figure 4.3: Comparison among precision, recall, and f1-score for class 1 (Exclamatory Sentence)	27
Figure 4.4: Comparison among precision, recall, and f1-score for class 2 (Assertive Sentence)	28

LIST OF TABLES

TABLES	PAGE NO
Table 1: Interrogative sentences data collection format	10
Table 2: Exclamatory sentences data collection format	11
Table 3: Assertive sentences data collection format	11
Table 4: Measure precision, recall, and f1-score for three classes (Naïve Bayes algorithm)	23
Table 5: Measure precision, recall, and f1-score for three classes (random forest algorithm)	24
Table 6: Measure precision, recall, and f1-score for three classes (decision tree algorithm)	24
Table 7: Measure precision, recall, and f1-score for three classes (support vector machine algorithm)	25
Table 8: K-Nearest Neighbor algorithm precision, recall, and f1-score	25
Table 9: Measure precision, recall, and f1-score for three classes (XGBoost algorithm)	26

CHAPTER 1

Introduction

1.1 Introduction

Language is the most effective means of expressing one's feelings, with a phrase serving as the basic unit of language text. A sentence is a group of words that together form a complete notion. From the inception to the present, in this modern world, the majority of work, including all research, publications, and equipment, has been done in English. For the creation of computer languages, extensive study is now being conducted on languages other than English. Bengali is the world's fifth most spoken mother tongue today. Furthermore, the language is the sixth most widely spoken in terms of total speakers. Researchers are presently studying many languages to see how they may be linked to machine learning and artificial intelligence. As a result, there has been a lot of work done recently in Bangladesh to include the mother tongue into the structural unique features and technology of the Bengali language. In this study, the classification of Bengali sentences has begun as a result of this inspiration. The act of categorizing a sentence into a preset category is known as sentence classification. In English, for example, Bengali sentences are classified according to their utility, where a functional statement refers to a speaker's goal. In Bengali, there are five different sorts of phrases that are evaluated based on their efficacy or meaning. Our datasets were divided into three categories of functional sentence types: introspective, interrogative, and exclamatory. These three types of sentences are extremely dissimilar to one another. To determine classification skills, multiple machine learning (ML) methods were applied to the random forest, nave base, decision tree classifier, SVM, KNN, and XGB classifier datasets. Our manual dataset was used as a forecast after this analysis. It was discovered that all of the models used yielded adequate results for such datasets. Random forest models, on the other hand, received the maximum score of 75.38 percent for perfect forecasting.

1.2 Motivation

Lately, NLP is one of the most popular and interesting fields of Artificial Intelligence. As a branch of AI, it helps computers as well as smart devices, understand, interpret and manipulate human languages. Moreover, NLP endeavors to fill the gap between computer understanding and human communication [19]. This behavior of NLP increased both its necessity and curiosity in AI sectors. With constant innovation and exploration is going on in this field, it is anticipated to grow in the oncoming day. According to a global market prediction, NLP revenue will grow from \$10.2 billion in 2019 to \$26.4 billion in 2024, with a Compound Annual Growth Rate (CAGR) of 21.0 percent throughout the forecast period [20]. The usage of huge amounts of smart technologies and devices, cloud-based solutions, and the thirst for AI-based text-sensing capable and understandable computers is the major growth factor of NLP. In Figure 1.2, It is supposed that NLP can be applied to any kind of AI field. It also refers that NLP is a powerful tool for ML and DL.

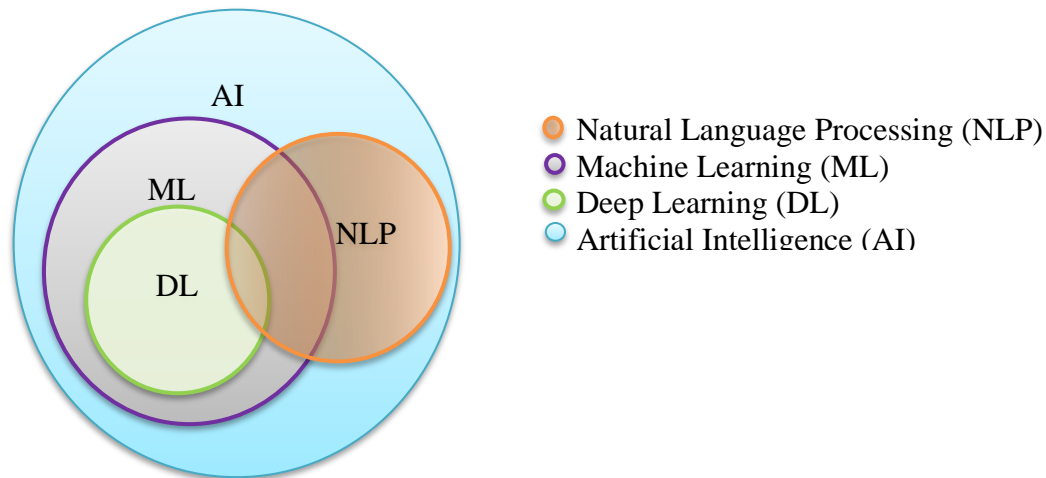


Figure 1.2: Fields of Artificial Intelligence

The key to doing something is its necessity, stakes, and possibilities offered that motivate a person. However, the demand and the scopes of the fields motivated us to work on it. On the other hand, researchers are working with different types of languages like English, Thai, Chinese, Japanese, Tamil, and so on. They are constantly looking for ways to teach computers their language. Inspired by this, we also decided to work in this field to introduce

©Daffodil International University

our mother tongue Bengali to computers. To enrich the field with Bengali, we have initially worked with Bengali Grammar analysis.

1.3 Research Objectives

One of the challenging fields in AI is NLP since it deals with human language. It is extremely diversified and can be spoken in a lot of ways. NLP Algorithms are used to implement these functionalities. Thus, is not a very easy implementation which is a component of text mining. Since unstructured data cannot be used for data retrieval, various methods are applied to text mining such as summarization, classification, clustering, information extraction, and visualization which are the categories under text mining. In this study, we work on the classification category of text mining. We have classified Bengali functional Sentence. From the beginning to the end of the work, several problems have been faced that we accepted as a challenge. The problems encountered are as follows:

- Finding out the scopes from the prior related works.
- The Dataset collection and preparation.
- Selecting and installation ML classification algorithms.
- Obtaining the expected outcome.
- During Analyze the result for multiclass.

However, overcome of those challenges are the objectives of this research. With the above problems in mind, we outlined the objectives of our study as follows:

- To track down the scopes from reviewing the works of literature which are done by prominent researchers.
- To collect relevant data and to develop the dataset. Work cannot be carried out without developing datasets.
- To develop the selecting algorithms for classification problems. Since the heart of the research is the algorithms. Without strong algorithms, the outcome won't be gained properly.

- To analyze the algorithms' performances for identifying the best model.

1.4 Report Layout

The following five chapters of this research paper have been examined: "Introduction," "Related Works," "Research Methodology," "Result Analysis," and "Conclusion and Future Works."

Chapter-1:

The first chapter covers the study's introduction. We explained the motivation for the research, the paper's objectives, and the report layout in this introduction.

Chapter-2:

In chapter two, related works are reviewed. In this review, we have focused on the former circumstances of the related researches which are done by the prominent researchers, and discussed the scope and the challenges of the research.

Chapter-3:

Chapter three conveyed the methodology of the study. Data collection and preprocessing, models' selection and description of the proposed models, and analyzing parameters were represented in this chapter.

Chapter-4:

Chapter 4 is about the experiment analyzing part. By this analysis, we picked out the best algorithm among the applying models. In our analyzing part, we considered data tables and graph figures to better understand and also compared the output data of the models among each other by those graphs and tables.

Chapter-5:

Chapter 5 is representing the summary, limitations, and future works of this research.

CHAPTER 2

Related Works

2.1 Related works

Thanyarat Nomponkrang et al. [1] suggested a module to categorize Thai sentences to determine their functionality as well as the relevant qualities and features of Thai sentences based on an algorithm. The module's major features are the term binary (TB) of the original phrase and the word frequency (TF) of the section of the speech. With the original phrase and POS word TF-IDF, it was proposed to use the SVM algorithm as the best model when comparing the two four classification algorithms. Quotes and greetings in sentences, on the other hand, were not taken into consideration in this article.

Chengwei Gu et al. [2] designed Natural Language Processing (NLP) for the classification of sentences in Chinese. The traditional method of machine learning was used. A fancy CNN architecture was applied to the classification of Chinese sentences. To improve, they have embedded linear SVMs. Also, *tanh* was enabled instead of ReLU and the results of the CNN model were improved for the work of Chinese sentence classification.

Cheng Liu et al. [3] proposed English standard-related text classification. The essential work of the paper was to mine a lot of data mining, stratification, and research in English text. In this study, the authors proposed a classification model based on cyclic neural networks. They applied due to the loop formation of the recurrent neural network (RNN). The accuracy of the RNN classification was better than the conventional network neural algorithm. Accuracy has reached about 96% with quality classification in a cyclical network.

According to Rizwan Ali Naqvi et al. [4], due to the complexity of the language, Roman Urdu news headlines were used as datasets. They classified the news into five sections. For classification, ML algorithms such as Long Short-Term Memory (LSTM), Multinomial Nev Base (MNB), Logistic Regression (LR), and Convolutional Neural Network (CNN)

were used to compare results. After applying a variety of models, the authors acknowledged that the MNB algorithm has been selected as the best perfection of 90.17%. In addition, rule-based procedures were employed to eliminate lexical distinctions.

Fuchun Peng et al. [5] Introduced to compare different methods of ML on Asian language text classification. Japanese and Chinese languages were considered where information about the scope of words in the written text was not available. It used the highest entropy model, Naive Bayes (NB), language modeling (LM), and support vector machine (SVM) methods. For both languages, the LM classifier offered better accuracy than the others. For the Chinese character and word level, LM gave about 86.7% and 89.2%, respectively, and for the Japanese character level about 84%. But SVM did not perform at the Japanese character level. The relationship between segmentation accuracy and word classification performance was also investigated.

Ping Zhang et al. [6] raised a research paper using machine learning models on text classification. In that case, three general text classification algorithms: K-Nearest Neighbor (KNN), Naive Bayes (NB), and Support Vector Machines (SVM) were introduced, all of which have good classification effects. After comparison, the experimental results indicated that the SVM model and the NB model provided better results.

2.2 Scopes of the Work

Reviewing the literature above, it is seen that work has been done to include different languages in machine learning. In addition, work has been done on the Bengali language, the amount of which is limited. For this reason, we have worked on the Bengali language to enrich the influence of the Bengali language in NLP i.e. machine learning in a bigger way. Although we classify sentences as a preliminary stage, this study will serve as a door opener for later work. Moreover, it is conceivable from the Literature Review that although work was done in other languages on this subject, not much work was done in Bengali. By

reviewing our research, new researchers on this subject will get the idea of working like word classification, prediction, structural classification, and prediction of Bangla sentences. On the other hand, it has been found that most of the previous works are Deep Learning-based works. As a result, those who want to work with a pure machine learning-based model on the Bengali language can also take ideas from this research paper.

2.3 Challenges

We have had to face some challenges as there is no work on the Bangla language related to this research. Out of all the challenges, the most difficult ones are stated below:

- Literature reviewing is the most important and prerequisite for any research. Which was a tough challenge for us as we could not find much work on this subject.
- The most challenging work was to collect data. As the work is the first attempt, we failed to collect online data set from many platforms such as Kaggol.
- Another challenge is to develop that dataset.
- We also faced the challenge of selecting and developing ML algorithms for classification.

CHAPTER 3

Research Methodology

3.1 Introduction

The work method was done step by step to find the best model for the classification of Bangla functional sentences through the machine learning method. As a result, the experiments followed a meaningful framework. A flow diagram for smoothing was built after selecting the ML algorithm. However, before selecting a model, the collected datasets were analyzed. Furthermore, six models were taken into consideration for analysis, described in Figure 3.1 as DT (Decision Tree), NB (Naive Bayes), SVM (Support Vector Machines), RF (Random Forest), KNN (K-nearest Neighbor), and XGB (XGBoost) respectively.

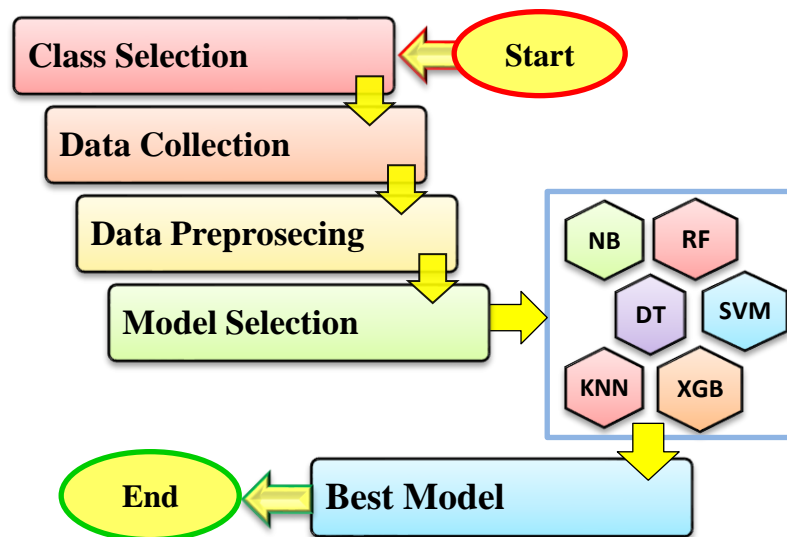


Figure 3.1: The Flow Chart of the Proposed Methodology

3.2 Class Selection

Labels or categories that are used to compare something are called classes. These are the classes that are produced as a result of classification work. There are four different sorts of classification that can be encountered when doing classification work. Multiple

classifications have been selected from these four. A multiclass classification is a classification that has more than two class labels. The class in this project was chosen from a variety of functional sentences in Bengali. The research was carried out with the help of the three most commonly used Bengali sentences. Three types of sentences exist: assertive, interrogative, and exclamatory. The classes are strongly formatted as interrogative (0), exclamatory (1), and assertive (2).

3.3 Data Collection Process

Although raw data is useful for getting started, it cannot be used in machine learning algorithms. Although it is possible to accomplish things in this manner, the best results are obtained by going a long way. As a result, it's critical to process the unprocessed data. To create a dataset, follow the steps outlined in Figure 3.3.1.



Figure 3.3.1: Diagram of Data Collection

The three common sources of data collection are Open source, internet, and artificial data generation. From the open sources, Data has been collected. All data were assembled from various Bengali poems, short story books, plays, novels, newspapers, articles, and web portals through a manual effort. In the first stage of data collection, all data was deposited on a Google sheet. After the collection was completed, the data set was sent to the next level for preprocessing.

A total of 7500 sentences [22] have been included as datasets at various intervals. Initially, 3000 datasets were included in the dataset. Formerly 3000 data points were collected once

more in the second step. Finally, 500 is added for each type of sentence of 1500 sum to complete the margin. In Figure 3.3.2, a pie chart is represented to show the percentage of each type of sentence.

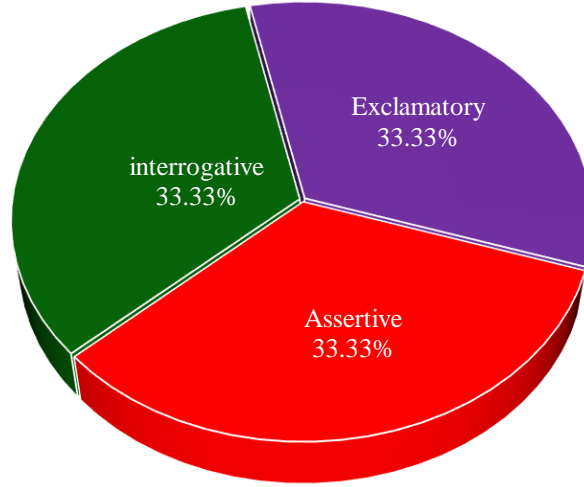


Figure 3.3.2: Pie Chart View of Collected Data

Data were organized into three categories when it was stored: context (sentence), context type (sentence name), and a class of Context. TABLE 1, TABLE 2, and TABLE 3 presented data collection examples of interrogative, exclamatory, and assertive sentences that are shown below.

TABLE 1: INTERROGATIVE SENTENCES DATA COLLECTION FORMATE

Context	Context_name	Class
তোমার নাম কি?	প্রশ্নবোধক বাক্য	0
কোনটি সবচেয়ে প্রাচীন সভ্যতা?	প্রশ্নবোধক বাক্য	0
ইতিহাসের প্রথম লিখিত আইন প্রণেতা কে?	প্রশ্নবোধক বাক্য	0

TABLE 2: EXCLAMATORY SENTENCES DATA COLLECTION FORMATE

Context	Context_name	Class
কত সুন্দর প্রকৃতি	বিস্ময়সূচক বাক্য	1
যদি আমরা জয়ী হতে পারতাম!	বিস্ময়সূচক বাক্য	1
কি সুন্দর নাম তোমার!	বিস্ময়সূচক বাক্য	1

TABLE 3: ASSERTIVE SENTENCES DATA COLLECTION FORMATE

Context	Context_name	Class
তার আদর্শ অবিস্মরণীয়।	বিত্তিমূলক বাক্য	2
তুমি অন্যায় কাজ করেছো।	বিত্তিমূলক বাক্য	2
স্কুলে শিক্ষাপ্রদানের ভাষা বাংলা।	বিত্তিমূলক বাক্য	2

3.4 Data Preprocessing

Data preparation is the first and most important stage in building a machine learning model. While data from the actual world carries many inconsistencies, incomplete and errors, or extroverts, it is not possible to create a model by manipulating that data manually. On the other hand, this study made use of a one-of-a-kind dataset. This is since it includes a manually created dataset that was previously less likely to be used. Accordingly, it must be needed equipped dataset for the machine learning model. At this stage, the dataset was

©Daffodil International University

prepared using a few steps propping different Python libraries. In Figure 3.4, the diagram is a depiction of the preprocessing approach.

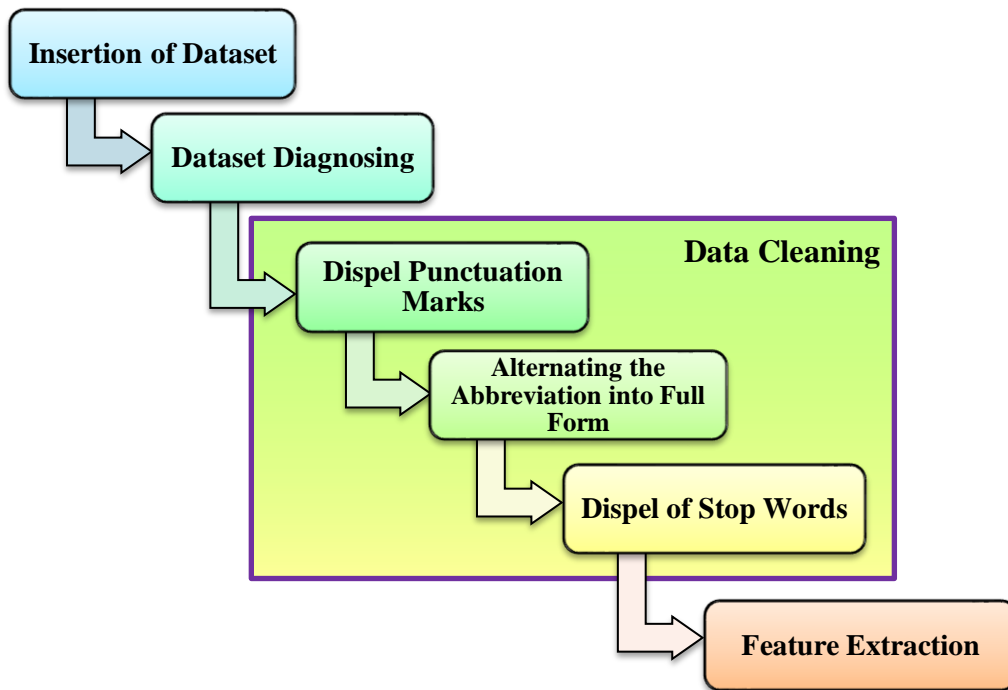


Figure 3.4: Diagram of Data Preprocessing

3.4.1 Insertion of Dataset

The data insertion step is very significant. This is because the model does not work at all when the collected data is inserted into any type of file format. Models work by inserting certain types of file formats such as CSV, or HTML, or XLSX file formats and move on to the next process for processing that file. The CSV file format of the collected dataset was adopted in our study. The Python library needed for further processing was also imported.

3.4.2 Dataset Diagnosing

This step refers to one of the ways of choosing valid data. Sometimes it is also called data diagnosis which is very essential in algorithms. In this plod, diverse types of scrutinizing

have been accomplished. After inserting the data, the data type of the dataset is first verified in this step. We got context as object type and class as integer type of our dataset which was a favorable aspect for us. In this plaid, various types of verification and selection have been done. After inserting the data, the data type of the dataset is first verified in this step. We got classes as context and integer as objects of our dataset which was a favorable aspect for us. The dataset is then monitored to see if there are any null values. To monitor and find these null values, *info()* function and *isna().sum()* functions are applied. We have also verified the data mean value, standard deviation, minimum and maximum values through the *describe()* function. All the applied functions are built-in functions of python. Finally, by checking and removing duplicate entries the dataset was made unique for ameliorating accuracy.

3.4.3 Dataset Cleaning

The step belongs to data cleaning of data preprocessing. Unnessecery things such as punctuation marks (*e.g.- “/”, “,”, “;”, “?”, “-”, “:” etc*), stop-words(*all Bengali connective words, e.g.- “যে”, “যেহেতু”, “আর”, “তারপর”, “তবুও”, “যদি”, “কিন্তু” etc*) are dispelled for feature extraction. Mainly, punctuation marks, stop-words do not carry much information rather it sometimes misguides the key feature. In addition, punctuation can make noise in the sentence. Here, the *NLTK* library was applied for tokenization, normalization, and vectorization before feature extraction. Splitting an entire sentence into words is known as tokenizing where a simple separator can be utilized for this. However, abbreviations are separated by “.”(*e.g.- বি.দ্রো.=“বিশেষ দ্রোষ্টব্য”, ডা. = “ডাক্তার” etc.*), the separator fails to split them. Consequently, the Abbreviations were alternated into full form. After that, punctuation marks and stop words are dispelled by read-through each token from the corpus. Since *Word-Tokenize* is a very popular function for tokenization and cleaning, we utilized this in our study to succeed in this stage.

3.4.4 Feature Extraction

Since ML algorithms can't understand the normal form of data, data needs to encode as integers i.e., the numeric form that generates the feature vectors. Vectorizing is a process of this encoding. Count Vectorizer or Bag of Words (BoW) commonly applies for vectorizing that reveals the existence of words in the data. It produces the result in binary. When a word is present in the text data it returns 1 for this word else returns 0 and creates a vector-matrix count in every single data. On the other hand, this vector conversion step is essential to fit the data into the algorithms. Hereafter, the vector-matrix dataset is split into two parts in an 85:15 ratio.

3.5 Machine Learning Model Selection

For classification, On the basis of our datasets, six machine learning models were deployed to forecast our findings. The whole dataset was splatted into two parts in the ratio of 85:15. In this research, 85% of data has been kept as a training set and 15% as a testing set. The applied algorithms are elucidated as follows:

3.5.1 Naive Bayes (NB)

It is a classification method in which a unique trait in a class stands out from the rest of the group. For really big datasets, this classifier is simple to use, fast, and effective. The method is also recognized as a highly advanced categorization tool. It has applications in real-time prediction, multiclass classification, and sentiment or text analysis. In this study, multinomial Naive Bayes was used [7]. Naïve Bayes based on Bayes Theorem. Baye theorem calculates conditional probability. Conditional probability measures the probability of a case occurring given that another case has occurred. The formula of Bayes Theorem is as follows-

$$P(X/Y) = \frac{P(Y|X).P(X)}{P(Y)} \quad (1)$$

In equation (1), X represents the class that means X tells us the sentence is interrogative or exclamatory or assertive sentence based on the given condition. Y represents the features that map the sentences where $Y = (Y_1, Y_2, Y_3, \dots, Y_n)$.

Now, The formula can be written like equation (2),

$$P(X/Y_1, Y_2, \dots, Y_n) = \frac{P(Y_1|X)P(Y_2|X)\dots P(Y_n|X)P(X)}{P(Y_1)P(Y_2)\dots P(Y_n)} \quad (2)$$

Therefore, the probability can be written as equation (3),

$$P(X/Y_1, Y_2, \dots, Y_n) \propto P(X) \prod_{i=1}^n P(Y_i|X) \quad (3)$$

However, in the case of us, we use naive Bayes to classify our sentences. Here, X will provide the maximum probability value. Thus, we can represent our maximum probability by equation (4) [15].

$$X = \operatorname{argmax}_x P(X) \prod_{i=1}^n P(Y_i|X) \quad (4)$$

3.5.2 Random Forest (RF)

The random forest is one of the most basic and straightforward models. Both calculation and regression analysis can be done with it. This is the most used method due to its adaptability, and it usually produces excellent results without the need for hyper-parameter customization. It's also part of a supervised machine learning method. The tree is now an ensemble as a result of this decision. Then it gives each of them a prophecy. It enhances the best tree to receive the most votes [8]. Simply put, the model generates and assembles numerous trees before arriving at acceptable and most viable outputs. The huge number of trees also prevents the overfitting problem. In the training phase, random forests contain many decision trees. For each decision tree, random forest calculates the importance of a node using Gini importance. Equation (3) is the calculation term of Importance of i^{th} number Node NI_i .

$$NI_i = W_i C_i - W_{L(i)} C_{L(i)} - W_{R(i)} C_{R(i)} \quad (5)$$

On a decision tree, the importance of a feature calculates by the following equation (6) where the summation of all node importance is divided by the total number of nodes. It can normalize a value between 0 and 1. The normalized formula is represented by equation (7).

$$FI_j = \frac{\sum_{i:\text{node } i \text{ splite on feature } j} NI_i}{\sum_{k \in \text{all nodes}} NI_k} \quad (6)$$

$$NORMFI_j = \frac{FI_j}{\sum_{i \in \text{all feature}} FI_i} \quad (7)$$

After that, random forest averages all the normalized values of decision trees. The summation of features importance for each tree is divided by the total number of trees which is represented by equation (8) [16].

$$RFFI_j = \frac{\sum_{i \in \text{all trees}} NORMFI_{ji}}{T} \quad (8)$$

3.5.3 Decision Tree (DT)

Decision Trees are used for classification and regression problems and are associated with supervised machine learning. It mostly provides data in the form of a tree. A decision tree predicts a record's class label by starting at the root of the tree. The root attribute and the record attribute are then compared. The comparison is completed, and the node jumps to the next node [9]. These algorithms are capable of working with both numerical and categorical data. The SOP (Sum of Products) norm, on the other hand, is upheld. In this algorithm, the following criteria calculate the values for each attribute. The attribute is placed on the root with a high value. First, the entropy was measured that measures the randomness of information being processed. The mathematical representation of entropy is as follows by the equation (10).

$$E(S) = \sum_{i=1}^c -p_i \log_2 p_i \quad (10)$$

For multiple attributes, the equation (11) is represented,

$$E(T, X) = \sum_{c \in E} P(c)E(c) \quad (11)$$

Now, calculate the information gain. The decrease of entropy is information gain. It calculates the difference in entropy between the before and after split average entropy. The mathematical term can be represented by equation (12) where T is the current state and X is the selected attributes. The simpler conclusion of information gain can be represented by equation (14).

$$IG(T, X) = E(T) - E(T, X) \quad (12)$$

$$IG = E(before) - \sum_{j=1}^k E(j, After) \quad (14)$$

Then, the Gini index is calculated. The equation (15) is for the Gini index calculation.

$$Gini = 1 - \sum_{i=1}^c (p_i)^2 \quad (15)$$

If we divide Information Gain(i.e. eq.14) by Split Information, Gain Ratio equation (16) will be got. The problem with IG by taking into account the number of branches is scalped by the gain ratio that would result before erection the split. IG is fixed by adopting the intrinsic information of a split into account [17].

$$Gain\ Ratio = \frac{IG}{\sum_{i=1}^k w_j \log_2 w_j} \quad (16)$$

3.5.4 Support Vector Machines (SVM)

Support Vector Machine (SVM) is a supervised machine learning technique that belongs to the SVM family. It is mostly used to solve categorization challenges. The models can also be employed in regression problems. The SVM algorithm's purpose is to build optimal decision boundaries that may split n-dimensional spaces in a class, with each item's value defining a distinct coordinate. This is referred to as the best decision boundary because it is a hyper-plane that can effectively separate two classes, and we use it to perform classification. The hyperplane's dimensions are determined by the number of features [10].

3.5.5 K-nearest Neighbor (KNN)

The K-nearest neighbor is a well-known non-parametric supervised technique. It can be used to solve problems like classification and regression. The categorization problem, however, is the most widely employed. A lazy learning model is also dubbed by KNN. It works quite slowly since it saves rather than learns data from its training dataset. When dealing with a large data set, this is a challenging condition to navigate. It predicts outcomes based on the data point or neighbor that is closest to the user [11]. This method divides fresh data into pieces that are more comparable to the new data as it is being trained. In this algorithm, we had needed to find out a function $h: X \rightarrow Y$, where KNN was an unknown observation x and $h(x)$ predict the output y .

First, the Euclidean Distance matrix is used (eq.17).

$$d(x, x') = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + \dots + (x_n - x_n')^2} \quad (17)$$

At that time, with the largest probability x is assigned to the class,

$$P(y = j / X = x) = \frac{1}{K} \sum_{i \in A} I(y^{(i)} = j) \quad (18)$$

In the KNN, K is a hyperparameter [18].

3.5.6 Extreme Gradient Boosting (XGB)

One of the best redacting algorithms is utilized for supervised learning is XGB (Extreme Gradient Boosting). Most data scientists prefer XGB whereas it has a high accomplished speed out of the core estimation. XGB is a machine learning technique based on decision trees that uses a gradient boost decision tree design. This speeds up the algorithm and improves its performance. XGB is used to tackle regression, classification, and prediction issues. [12].

3.6 Cross-Validation

Cross-validation is a method of data resampling to evaluate the generalizability of predictive models and to prevent overfitting. Among many of the resampling methods, cross-validation is customarily exerted to attune model parameters [13]. Because of its simplicity, K-fold cross-validation is a common method. It assumes that there is an opportunity to attend the training and test set of each observation from the original dataset. It is only used in a predetermined dataset. k is a single parameter in k-fold, which signifies that the given dataset is separated into groups. We used 10-times cross-validation for our datasets. For our dataset, we employed 10-fold cross-validation. By Figure 3.6 this scheme for k=10 is expounded, i.e., 10-fold cross-validation.

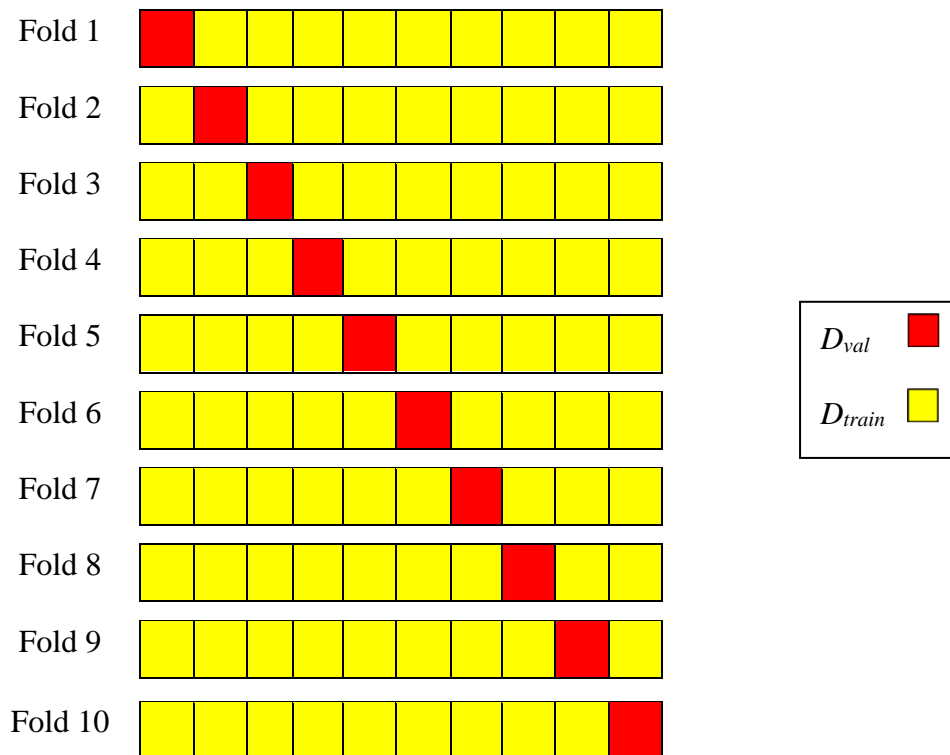


Figure 3.6: K-Fold Cross Validation where k=10 and the data set are randomly split into 10 detach subsets.

The first subset serves as a validation set D_{val-1} , while the following nine subsets serve as a training set $D_{train-1}$ in the first fold. Similarly, the validation set D_{val-2} is the second subset

of the second fold, whereas the remaining subsets constitute the training set $D_{train-2}$. This method of working has been revolved unless each k subset acts as a validity set.

3.7 Performance Parameters

3.7.1 Confusion Matrix

A confusion matrix is a metric for determining how well a categorization system performs. It's a table that combines four separate parameters with projected and real-world values. True positive (TP), false negative (FN), true negative (TN), and false-positive (FP) are the four metrics used to assess it [14]. The exploratory outcome may be conveniently concise by the following Figure 3.7.1 [21]:

		(Real)	
		Positive	Negative
(Prediction)	Positive	TP	FP
	Negative	FN	TN

Figure 3.7.1: Confusion Matrix

According to this figure 4, Precision, Recall, Accuracy can be measured. After calculating the precision and recall, using the measuring values of those, F1-score can be calculated.

3.7.2 Precision

Precision is defined as the ratio of true positives to total positives (true positives and false positives). A positive anticipated value is another term for this. The term of precision p is represented by the equation (19):

$$p = \frac{TP}{TP+FP} \quad (19)$$

3.7.3 Recall

The recall of a classifier refers to its capacity to intuitively locate all positive samples. The lowest and highest values are 0 and 1, respectively. In a word, recall is a measure of positive identification of truth. The term of recall r is as follows:

$$r = \frac{TP}{TP+FN} \quad (20)$$

3.7.4 F1-score

We can't easily compare two models so we use f1-scores to compare them. The F1-score is measured by the harmonic mean of recall and precision. So, the formula of F1-score will be as following equation (21).

$$F1\text{-score} = 2 \times \frac{p \cdot r}{p+r} \quad (21)$$

3.7.5 Accuracy

Accuracy is a metric for determining classification models. In a family way, accuracy is the fraction of prophecies the model got accurate. Formally, accuracy has the following equation (22):

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{TP+TN}{TP+FP+TN+FN} \quad (22)$$

CHAPTER 4

Result Analysis

Our study's mold began with 3000 data points. When working with that dataset, RF provided better accuracy results than the others. The accuracy stated was 62 percent. When the data was expanded to 6000, RF held a maximum value of 68.4 percent, with a 6.4 percent increase the accuracy. 7500 data has been added because it did not meet our expectations. After doing this, for the third time, it showed a pretty satisfactory result that we can afford. It was very noticeable that the average increment of each algorithm was 5% accuracy. Cross-validation was created for every single algorithm to achieve optimal results. However, the purchase was a bit of a letdown. The comparison of test accuracy and cross-validation score was presented with the help of Figure 4.1.

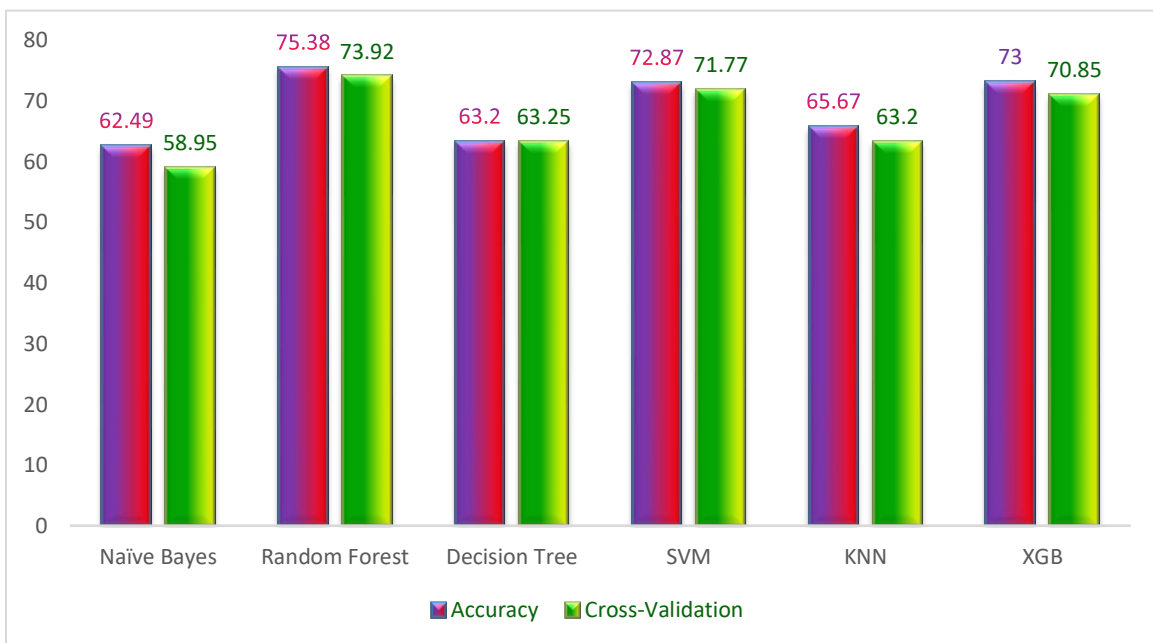


Figure 4.1: Comparison between test and cross-validation accuracy of different machine learning techniques

For NB, RF, SVC, KNN, DT Classifier, and XGB Classifier, the proposed experiment accuracy rate in TABLE I was 62.49%, 75.38%, 72.87%, 65.67%, 63.2%, and 73% as well.

On the other hand, the Cross-validation score exposed in the table was 58.95%, 73.925%, 71.77%, 63.2%, 63.25%, and 70.85%. After comparison between accuracy and cross-validation showed that for NB, RF, SVC, KNN, and XGB Classifier there was a significant reduction of 3.54%, 1.46%, 1.1%, 2.47%, and 2.15% respectively. However, for the DT algorithm, a meager accretion of 0.05% was seen. Despite an exhaustive reduction of accuracy after Cross-validation, it was proved that RF had the best scenery of score with a 75.35% value of perfection.

To evaluate the performance of each categorization, precision, recall, F1-score, and support were surveyed. Differences in criteria were evident in this comparison. TABLE 4 shows a comparison of the classes that have been used. In the instance of NB, recall and F1-score propped up a very close score to each other for the interrogative or class 0 precision, with 358 as the support. However, for exclamatory or class 1, it was noticed that precision was less than class 0 with recall and the f1-score was adaptable with 405 support. On the other hand, for assertive or class 2, in the case of NB, each parameter was very bad excluding support.

TABLE 4: MEASURE PRECISION, RECALL, AND F1-SCORE FOR THREE CLASSES (NAÏVE BAYES ALGORITHM)

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
62.49	58.95	Interrogative	0.67	0.69	0.68
		Exclamatory	0.61	0.69	0.65
		Assertive	0.58	0.48	0.53

Precision, recall, and f1-score for Interrogative or class 0 were determined to be 0.79 and support to be 358 in TABLE 5. For class 1, precision and f1-score were decreased to 0.72 and 0.77, respectively. Furthermore, recall soared to 0.82 with the help of 405 support. All of the criteria in Class 2 declined except for precision.

TABLE 5: MEASURE PRECISION, RECALL, AND F1-SCORE FOR THREE CLASSES (RANDOM FOREST ALGORITHM)

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
75.38	73.92	Interrogative	0.79	0.79	0.79
		Exclamatory	0.72	0.82	0.77
		Assertive	0.75	0.64	0.69

TABLE 6 demonstrates that the Decision Tree technique had precision, recall, f1-score, and support of 0.65, 0.68, 0.66, and 488 for class 0. Precision, recall, and f1-score for class 1 were all 0.64, with 538 support. For class 2, the precision, recall, f1-score, and support scores were 0.60, 0.58, 0.59, and 478, respectively. TABLE 6 shows, on the other hand, that Test accuracy has increased marginally, with a value of 63.25. However, the increase was negligible.

TABLE 6: MEASURE PRECISION, RECALL, AND F1-SCORE FOR THREE CLASSES (DECISION TREE ALGORITHM)

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
63.2	63.25	Interrogative	0.65	0.68	0.66
		Exclamatory	0.64	0.64	0.64
		Assertive	0.60	0.58	0.59

TABLE 7 contains all of the support vector machine's parameters. Precision was 0.74, the recall was 0.74, f1-score was 0.74, and support for class 0 was 488. Precision, recall, f1-score, and support for class 1 were also found to be 0.71, 0.79, 0.75, and 538, respectively.

The precision of 0.76, recall of 0.67, f1-score of 0.71, and support of 474 were obtained for class 2.

TABLE 7: MEASURE PRECISION, RECALL, AND F1-SCORE FOR THREE CLASSES (SUPPORT VECTOR MACHINE ALGORITHM)

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
72.78	71.77	Interrogative	0.74	0.74	0.7
		Exclamatory	0.71	0.79	0.75
		Assertive	0.76	0.67	0.71

Precision was 0.59, the recall was 0.79, f1-score was 0.67, and support was 483 for class 0 in TABLE 8 for the K-Nearest Neighbors algorithm. Furthermore, the precision, recall, f1-score, and support for class 1 were 0.69, 0.68, 0.68, and 519, respectively. On the other hand, TABLE 8 shows a precision of 0.75, recall of 0.50, f1-score of 0.60, and support of 498 for class 2.

TABLE 8: K-NEAREST NEIGHBOR ALGORITHM PRECISION, RECALL, AND F1-SCORE

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
65.67	63.2	Interrogative	0.59	0.79	0.67
		Exclamatory	0.69	0.68	0.68
		Assertive	0.75	0.50	0.60

TABLE 9 represents precision as 0.74, recall as 0.77, f1-score as 0.67, and support as 483 for class 0 for the XGBoost classifiers. For class 1, however, precision, recall, f1-score, ©Daffodil International University

and support were 0.69, 0.68, 0.68, and 519, respectively. Furthermore, precision was 0.74, the recall was 0.60, f1-score was 0.70, and support was 498 for class 2.

TABLE 9. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (XGBOOST ALGORITHM)

Test Score	Cross-Validation Score	Class	Precision	Recall	F1-Score
73	70.85	Interrogative	0.74	0.77	0.75
		Exclamatory	0.71	0.77	0.74
		Assertive	0.74	0.66	0.70

To make the comparison clear, the graph was rendered. Figure 4.2 was outlined for class 0. In that graph, Random Forest was the highest-scoring model for precision, recall, and f1-score. Thus, the KNN model was displayed in that graph with a high recall score of 0.79. The chart is illustrated for Class 1 in Figure 4.3. Of the six classifiers, Random Forest again topped for three parameters. Additionally, Figure 4.4 was sketched for Class 2, indicating that SVC is the top model with the highest standard of parameters. Moreover, Random Forrest also scored very close to SVC. After all comparisons, it has been concluded that Random Forest, SVC, and XGB classifiers performed much better than Naive Bayes, Decision Tree, and KNN. Nevertheless, Random Forest provided the winning score in most of the comparisons.

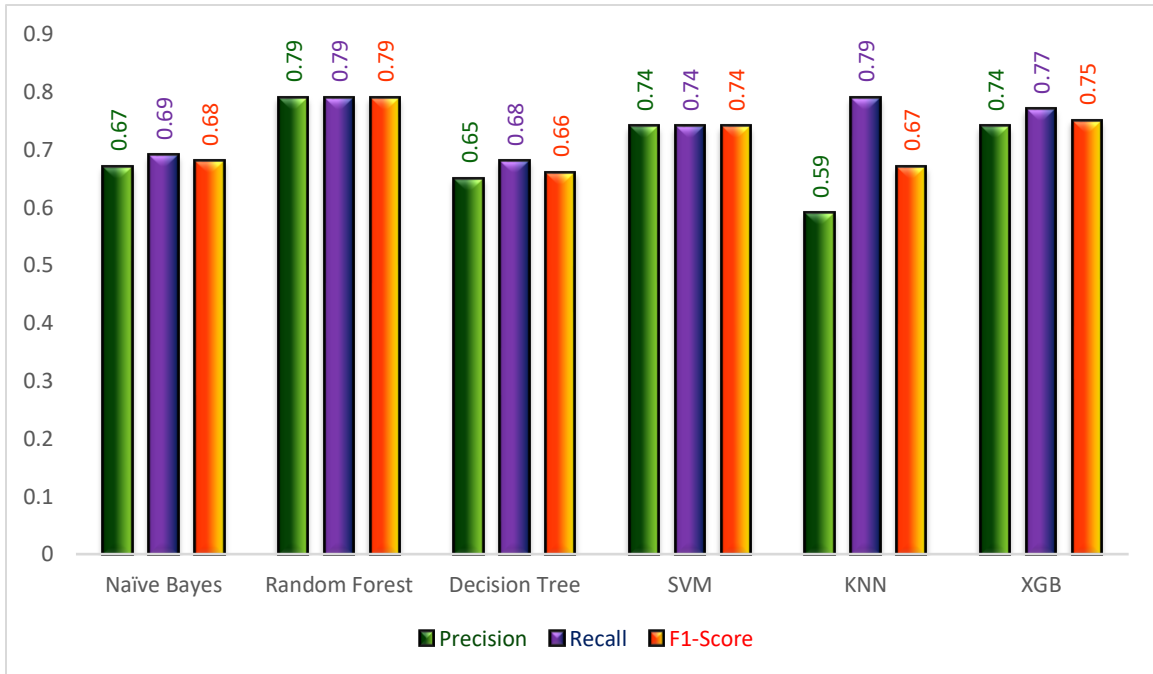


Figure 4.2: Comparison among precision, recall, and f1-score for class 0 (Interrogative Sentence)

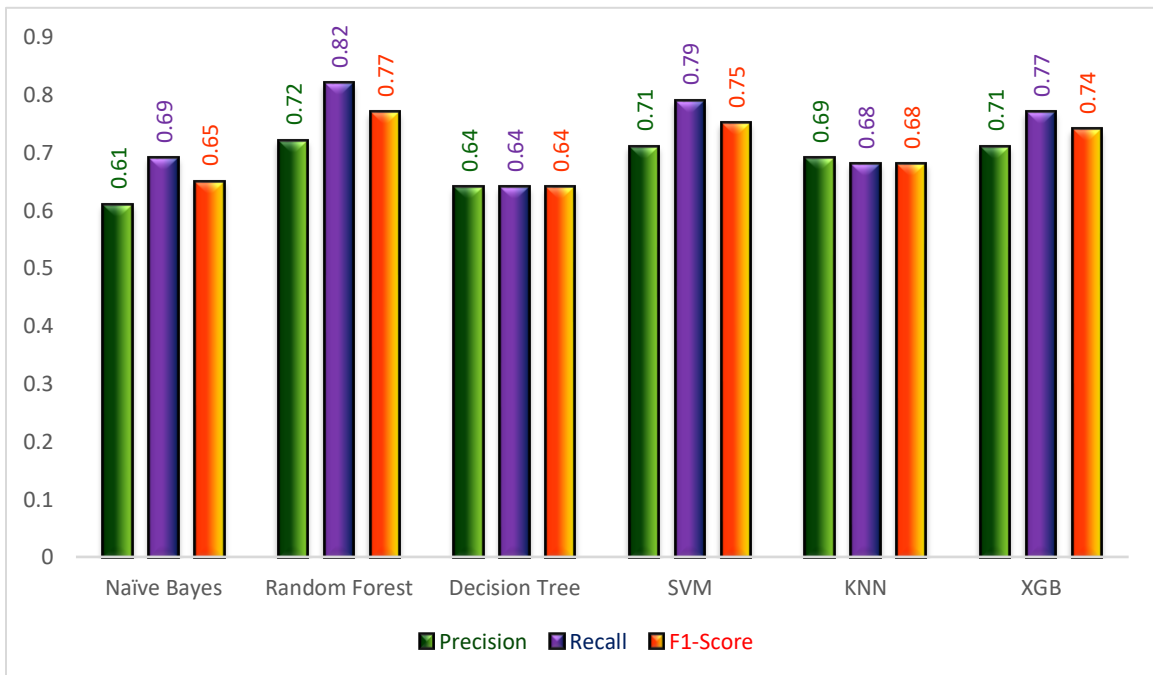


Figure 4.3: Comparison among precision, recall, and f1-score for class 1 (Exclamatory Sentence)

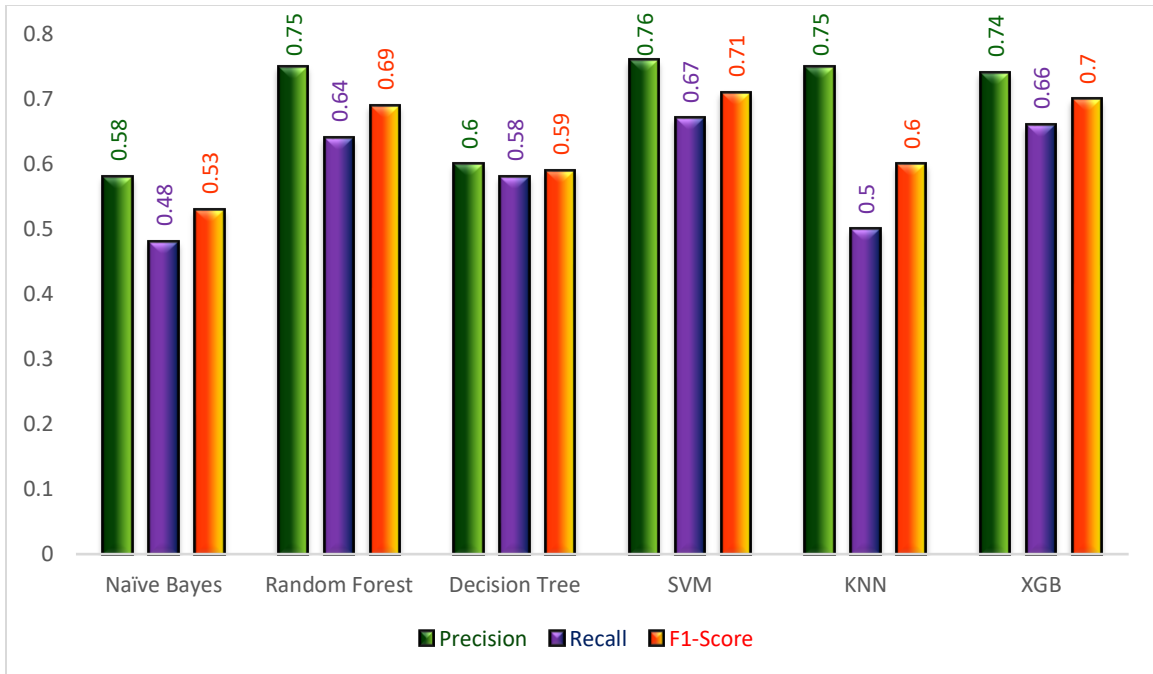


Figure 4.4: Comparison among precision, recall, and f1-score for class 2 (Assertive Sentence)

CHAPTER 5

Conclusion and Future Work

5.1 Conclusion

The paper offers comparisons between ML classifiers and seeks out the paramount methods for classifying Bengali sentences. Random Forest has surpassed 75.38% to classify different types of Bangla sentences with accuracy in this analysis. Thus, if the dataset can be extended by a large margin, the results will be more fruitful.

5.2 Future Works

As the work was just the beginning of our idea implementation. In the future, we have a plan that DL algorithms will be implemented in this dataset by adding the residual functional sentences as classes. Moreover, the recent world is being driven by dealing with online platforms. Educational works are also being converted into smart device-dependent. Keeping in mind this concept, we will try to develop a smart application for Bengali grammar learning where we will try to include more AI technological grammar learning modules along with this classification model concept, too. So that, a learner can able to learn the class of a sentence with other Bengali grammatical lessons very comfortably.

REFERENCES

- [1] T. Nomponkrang, the Department of Computer Education, Faculty of Technical Education, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, C. Sanrach, and the Department of Computer Education, Faculty of Technical Education, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, "The comparison of algorithms for Thai-sentence classification," *Int. J. Inf. Educ. Technol.*, vol. 6, no. 10, pp. 801–808, 2016.
- [2] C. Gu, M. Wu, and C. Zhang, "Chinese sentence classification based on convolutional neural network," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 261, p. 012008, 2017.
- [3] C. Liu and X. Wang, "Quality-related English text classification based on recurrent neural network," *J. Vis. Commun. Image Represent.*, vol. 71, no. 102724, p. 102724, 2020.
- [4] M. Adnan Khan, R. Ali Naqvi, N. Malik, S. Saqib, T. Alyas, and D. Hussain, "Roman Urdu news headline classification empowered with machine learning," *Comput. mater. contin.*, vol. 65, no. 2, pp. 1221–1236, 2020.
- [5] F. Peng and X. Huang, "Machine learning for Asian language text classification," *J. Doc.*, vol. 63, no. 3, pp. 378–397, 2007.
- [6] P. Zhang and Y. Fang, "Research on text classification algorithm based on machine learning," *J. Phys. Conf. Ser.*, vol. 1624, p. 042010, 2020.
- [7] A. McCallum and K. Nigam, "A comparison of event models for naive bayes text classification," 1998, pp. 41–8.
- [8] M. Pal, "Random forest classifier for remote sensing classification," *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, 2005.
- [9] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Trans. Syst. Man Cybern.*, vol. 21, no. 3, pp. 660–674, 1991.

- [10] S. Tong and D. Koller, “Support vector machine active learning with applications to text classification,” *Journal of Machine Learning Research*, vol. 45–66, 2001.
- [11] A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, “k-Nearest Neighbor Classification,” in *Data Mining in Agriculture*, New York, NY: Springer New York, 2009, pp. 83–106.
- [12] N. Fazakis, G. Kostopoulos, S. Karlos, S. Kotsiantis, and K. Sgarbas, “Self-trained eXtreme gradient boosting trees,” in *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, 2019.
- [13] D. Berrar, “Cross-Validation,” in *Encyclopedia of Bioinformatics and Computational Biology*, Elsevier, 2019, pp. 542–545.
- [14] G. Forman, “An extensive empirical study of feature selection metrics for text classification,” *The Journal of Machine Learning Research*, vol. 3, pp. 1289–305, 2003.
- [15] H. Zhang and D. Li, “Naïve Bayes Text Classifier,” in *2007 IEEE International Conference on Granular Computing (GRC 2007)*, 2007.
- [16] J. M. Chen, “An introduction to machine learning for panel data: Decision trees, random forests, and other dendrological methods,” *SSRN Electron. J.*, 2020.
- [17] N. S. Chauhan, “Decision tree algorithm, explained,” *Kdnuggets.com*. [Online]. Available: <https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html>. [Accessed: 06-Nov-2021].
- [18] *Medium.com*. [Online]. Available: <https://medium.com/@rdhawan201455/knn-k-nearest-neighbour-algorithm-maths-behind-it-and-how-to-find-the-best-value-for-k-6ff5b0955e3d>. [Accessed: 06-Nov-2021].
- [19] “5 Natural Language Processing examples: How NLP is used,” *Bloomreach.com*. [Online]. Available: <https://www.bloomreach.com/en/blog/2019/09/natural-language-processing.html>. [Accessed: 06-Nov-2021].

- [20] “Natural Language Processing (NLP) market insights by emerging trends, future growth, revenue analysis, demand forecast to 2024,” *MarketWatch*, 26-Oct-2021. [Online]. Available: <https://www.marketwatch.com/press-release/natural-language-processing-nlp-market-insights-by-emerging-trends-future-growth-revenue-analysis-demand-forecast-to-2024-2021-10-26?tesla=y>. [Accessed: 06-Nov-2021].
- [21] C. Goutte and E. Gaussier, “A probabilistic interpretation of precision, recall and F-score, with implication for evaluation,” in *Lecture Notes in Computer Science*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 345–359.
- [22] *Sentence Classification Dataset: Google.com*. [Online]. Available: <https://drive.google.com/file/d/1arhxm5LfcGDJNYixbsMs508YZM>. [Accessed: 07-Nov-2021].
- [23] A. Biswas, M. Rahman, Z. J. Orin, and Z. Hasan, “Bengali functional sentence classification through machine learning approach,” in 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021.

APPENDIX

In this affiliation, the published paper of our research is adjoined as an appendix which has been published in a conference named “**THE 12th INTERNATIONAL CONFERENCE ON COMPUTING, COMMUNICATION AND NETWORKING TECHNOLOGIES (ICCCNT)**”, July 6-8 IIT-Kharagpur, India.

Bengali Functional Sentence Classification through Machine Learning Approach

Antara Biswas
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
antara15-10644@diu.edu.bd

Musfiqur Rahman
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
mushfiqur15-10872@diu.edu.bd

Zahura Jebin Orin
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
zahura15-11127@diu.edu.bd

Md Zahid Hasan
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
zahid.cse@diu.edu.bd

Abstract— In the early time, very few studies were accomplished in Bengali functional sentences. However, the study on Bengali has incredibly increased for its structural diversity. Inspired by those studies, Functional sentence classification in Bengali language was completed including machine learning approaches to classify the sentences. Three types of Bengali functional sentences such as Assertive, Interrogative and Exclamatory have been considered for the research. So the leading purpose of the study is to classify the sentence and find out the best algorithm with comparing accuracy rate. Data have been collected, categorized and processed the dataset properly to avoid the conflict. Some popular machine learning algorithms such as Naive Bayes (NB), Decision Tree Classifier (DT), SVM, KNN, Random Forest (RF), and XGB Classifier have been implemented to compare accuracy rates. Parameters such as Precision, Recall, F1-Score, Support and Confusion matrix have been calculated for the comparison. The comparison demonstrated that performance of the Random Forest, SVC, and XGB Classifier is better than Naive Bayes and Decision Tree Classifier. Remarkable issue is that the Random Forest algorithm provided the highest performance value with an accuracy of 75.38% which is average performance for such a dataset.

Keywords— *Functional Sentence, Assertive, Interrogative, Exclamatory, Multiclass, Naive Bayes, Random Forest, Decision Tree Classifier, SVM, KNN, XGB Classifier*

I. INTRODUCTION

Language is the best way to express someone's sense where a sentence is the textual unit of language. Actually, a sentence is a set of words that in principle tells a complete thought. From the beginning to now, in this technological world, most of the work, all research works, journals or devices are done following English language. Extensive research is currently being done on other languages than English for the development of computer languages. In the present world, Bengali is the fifth most-spoken native language. Moreover the language is the sixth most spoken by total number of speakers. Researchers are currently conducting various researches on different languages in order to connect languages with Machine Learning and Artificial Intelligence. Accordingly, for the structural special features of the Bengali language and to employ the mother tongue in technology, a lot of work is being done in Bangladesh recently. Stimulated by this, the work has started on the classification of Bengali sentences in this paper. Sentence

Classification is the task of classifying a sentence under a predefined category. As like English, there is the classification of Bengali sentences according to their functionality where a functional sentence means to the purpose of a speaker. In Bengali, five types of sentences are considered according to their functionality or meaning. Among these, our dataset was categorized into 3 types of functional sentence classes: Assertive, Interrogative and Exclamatory. These three types of sentences are quite different from each other. In this research, several machine learning(ML) algorithms: Random Forest, Naive Bayes, Decision Tree Classifier, SVM, KNN, XGB Classifier have been applied to the dataset to determine the efficiency of the classifiers. Analyzing this, our manual dataset was considered to predict. It was observed that all the applied models lodged a considerable result for such a dataset. However, Random Forest models gave the highest score 75.38% of the perfect forecast.

II. LITERATURE REVIEW

Thanyarat Nomponkrang et al. [2] proposed a module to extract the features of Thai sentences according to the function of the sentences and the appropriate features and algorithm to classify the Thai sentences. Term binary (TB) of key phrases and term frequency (TF) of part of speech were extracted as main features of the module. Comparing two 4 classification algorithms, the SVM algorithm was proposed as the optimal model that had the key phrase and TF-IDF term of POS. However, adages and greeting sentences were not considered in this paper.

Chengwei Gu et al. [3] designed Natural Language Processing (NLP) for sentence classification in the Chinese language. The traditional way of machine learning was used. A novel architecture of CNN was applied to Chinese sentence classification. For betterment, they also embedded linear SVM. Besides, tanh was activated instead of ReLU and the result of the CNN model improved for Chinese sentence classification tasks.

Cheng Liu et al. [4] proposed English quality-related text classification. The vital work of the paper was to mine, stratify and research great quantity data in English text. In this study, the authors propounded a categorization model based on Cyclic Neural Network. The Recurrent neural network

(RNN) was applied because of its loop structure. The accuracy of RNN classification was better than the conventional network neural algorithms. The accuracy with quality categorization reached about 96% in a cyclic network.

According to Rizwan Ali Naqvi et al. [5], Roman Urdu News Headlines were used as a dataset because of the complexity of this language. They had classified the news into five categories. For the classification, algorithms of machine learning like long short-term memory (LSTM), Multinomial Naïve Bayes (MNB), Logistic Regression (LR), and Convolutional Neural Network (CNN) were utilized to compare the perfection of results. After applying different types of models, the authors confessed that the MNB classifier is selected as the best perfection of 90.17%. In addition, rule-based techniques were used to remove the lexical variation.

Fuchun Peng et al. [6] introduced to compare several techniques of machine learning on text classification of Asian languages. Japanese and Chinese languages were considered where information on word circumference is unavailable in written text. It implemented the Maximum entropy model, language modeling (LM), Naive Bayes (NB), and Support vector machines (SVM) approaches. For both languages, the LM classifier gave better accuracy than others. For Chinese character and word level, LM gave about 86.7% and 89.2% respectively, and 84% for Japanese character level. But in the Japanese character level SVM didn't perform. The relationship between word classification performance and segmentation accuracy were also investigated.

Ping Zhang et al. [7] raised research on text classification applying machine learning models. In that case, three typical text classification algorithms: Naive Bayes (NB), K-Nearest Neighbor (KNN) and Support Vector Machines (SVM) were introduced and all of which have good classification effects. After comparison, the experimental results indicated that the NB model and SVM model had provided good results.

However, observing the previous literature, most of the operation focused on NLP based solutions for classifying the sentences. Some of the authors have illustrated the machine learning algorithm to classify the sentences but using other languages except the Bengali. Therefore, the main scheme of the study is to classify the Bengali sentences through traditional machine learning approaches only.

III. PROPOSED METHODOLOGY

Due to finding out the best model for Bengali Functional Sentence Classification by machine learning approaches the working procedure was done following step to step. Accordingly, a contextual framework was followed in the experiments. After selecting ML algorithms, A flow diagram was created for smoothness. However, before the model selection the collected dataset was processed. Additionally, six models were considered for the analysis which are mentioned in fig. 1 as NB(Naive Bayes), RF(Random Forest), DT(Decision Tree), SVM(Support Vector Machines), KNN(K-nearest Neighbor) and XGB(XGBoost) respectively.

A. Select Class

The labels or categories which are done for comparing something is termed as Class. These classes are used as the output result of classification tasks. For classification tasks four types of class may be encountered. Among these four, multi-class classification has been selected. The classification task which has more than two class labels is known as multi-class classification. In this work classes have been selected from different functional sentences of Bengali language. The work has been done with three most frequently used types of sentences in Bengali language. These are Assertive, Interrogative and Exclamatory respectively. Here classes have been formatted assertive as 2, exclamatory as 1 and lastly interrogative as 0.

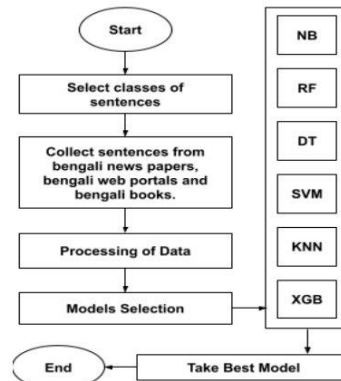


Fig. 1 The flow chart of the proposed methodology

B. Data Collection Process

To start the work raw data is excellent but it can't be inserted into machine learning algorithms. Though it can be done this way, the most desired outcome remains far away. Due to such reasons it is very much essential to process raw data. The steps have to be followed as below for building a dataset:

Collecting → Preprocessing

Open source, internet and artificial Data Generation are the three usual sources of data collection. Data has been collected from open source. All the data was collected by manual effort from various Bengali poems, dramas, novels, short story books, articles, newspapers and web portals as well. In total 7500 sentences [16] have been inducted as a dataset in different intervals. At first 3000 data was included in the dataset. Then in the second phase again 3000 data was adopted. Finally 500 for each type of sentence in sum 1500 more was added for completing the margin. While storing the data, it was classified into three sections of context (sentence), context type (sentence name) and class of Context. Fig. 2, fig. 3 and fig. 4 represented the example of interrogative, exclamatory and assertive sentences which have been shown below as well.

Context	Context Name	Class
তোমার নাম কি?	প্রশ্নবোধক বাক্য	0
তুমি কোথায় থাকো?	প্রশ্নবোধক বাক্য	0
ইতিহাসের প্রথম লিখিত আইন প্রণেতা কে?	প্রশ্নবোধক বাক্য	0
বাংলাদেশের প্রাচীন জাতি কোনটি?	প্রশ্নবোধক বাক্য	0

Fig. 2: Example of Interrogative Sentences

Context	Context Name	Class
কি সুন্দর দৃশ্য!	বিস্ময়সূচক বাক্য	1
মরি! মরি! কি সুন্দর প্রভাতের সোভা!	বিস্ময়সূচক বাক্য	1
সর্বনাশ, এটা তুমি কি করলে!	বিস্ময়সূচক বাক্য	1
হুম! হুম! লোকটা অকালেই চলে গেল!	বিস্ময়সূচক বাক্য	1

Fig. 3: Example of Exclamatory Sentences

Context	Context Name	Class
পরোপকারীকে সবাই শ্রদ্ধা করে।	বিবৃতিমূলক বাক্য	2
সুখবরটা পেয়ে সে আনন্দ পেয়েছে।	বিবৃতিমূলক বাক্য	2
রাত্রি প্রভাত হলে পশিরা গান গেয়ে ওঠে।	বিবৃতিমূলক বাক্য	2
গুণী লোক বিনয়ী হয়।	বিবৃতিমূলক বাক্য	2

Fig. 4: Example of Assertive Sentences

C. Data Pre Processing

There is a unique dataset used in this research. Because it has a manually built in dataset with very poor probability of being used before. For the betterment of the research the data have been sorted and filtered thoroughly for any invalid result. But the result was very amusing with no invalid result. Cross checking was also done in search of any kind of null value. Then, the punctuation marks were dispelled. After that, the connective words such as “যে”, “যেহেতু”, “আর”, “তারপর”, “তবুও”, “যদি”, “কিন্তু” etc. were dispelled as stop words. Some contracted form of words like “বিবিশেষ দ্রষ্টব্য = .দ্র.”, “ডাডাক্তার = .” etc. were found in data which were abbreviated in due form for better figure. Besides, statistical reports of Dataset have been identified, too. Python Libraries were used in this case to process. The data type was String. Due to this, Python Libraries were used to convert the String value into numeric value for fitting it into the machine learning approaches. In fig. 5, the diagram is the presentation of the preprocessing procedure.

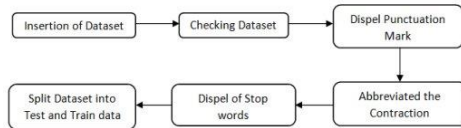


Fig. 5: Diagram of Data Preprocessing

D. Select Machine Learning Models

For the classification, six machine learning models were applied which predict our result based on our dataset. The whole dataset had been splitted into two parts in 85:15 ratio. One is the trained dataset which contains 85% data from the dataset and another one is the tested dataset which contains

the rest of 15%. The applied algorithm are elucidated as following:

1) *Naive Bayes (NB)*: It is a classification technique which assumes that an exceptional feature in a class is different from any other presence features. This classifier is easy to use and faster and very effective for very large datasets. It is also known that the method is a highly sophisticated classification method. It can be used for real time prediction, multiclass and text or sentiment analysis. In the research, multinomial naive bayes were executed[11].

2) *Random Forest (RF)*: One of the simplest and easiest models is Random Forest to use. It can be used for both calculation and regression analysis. It is the most famous algorithm for its flexibility and most of the time, it also provides a great result without hyper-parameter tuning. Besides, it belongs to a supervised machine learning algorithm. This makes an ensemble of decision trees. Then it predicts each of them. It becomes the best tree which gets the maximum votes[8]. Simply it can be said that the model makes multiple trees and makes them together before that gets the accurate and most performing result.

3) *Decision Tree (DT)*: Decision tree belongs to supervised machine learning and also used for classification and regression problems. It mainly represents its data as a tree. A decision tree works from its root of the tree for predicting a class label for a record. After that, it compares root attribute and record attribute. After completing comparison jumps to the next node[9]. These algorithms can work with both categorical and numerical data. On the other hand, it maintains the rules of Sum Of Products(SOP).

4) *Support Vector Machines (SVM)*: Support Vector Machine(SVM) is also a family member of supervised machine learning algorithms, too. Generally, it is mostly utilized in classification problems. Besides, the models can be used in regression challenges. The goal of the SVM algorithm is to make the best decision boundary that can separate n-dimensional space into classes, so the value of each item places a particular coordinate. It is called the best decision boundary as hyper-plane which can differentiate two classes very well and we perform classification by it. The dimension of a hyper-plane depends on the number of features[13].

5) *K-nearest Neighbor(KNN)*: K-nearest neighbor is a non-parametric supervised algorithm. It can be used for both classification and regression problems. But most of the use is noticed in classification problems. KNN is also called lazy learning algorithm. It performs very lazily because it doesn't learn data from it's training dataset instead it stores the dataset. It is a very tough situation when we use it with a very large data set. It predicts the result depending on the nearest neighbors or data points. When this algorithm trains new data, it classifies the new data into a category which is more similar to the new data.

6) *XGBoost(XGB)*: XGBoost is a decision tree based machine learning algorithm that is an implementation of gradient boost decision trees design. It makes the algorithm faster and increases the performance. To solve regression,

classification, ranking and user defined prediction problems XGBoost is performed.

E. Cross Validation(k-fold)

K-fold Cross validation is a very popular method because of its easy usage. It convenes that every observation from the main dataset has the chance of appearing in the training and test set. It is simply used in a limited dataset. In k-fold, k is a single parameter which means the groups of the given dataset are split. We have used 10-fold cross validation for our dataset.

F. Performance Parameters

a. Confusion matrix:

Confusion matrix is performance measurement of classification problems. This is a table which combines 4 different parameters of predicted and actual values. It measures by true positive, false negative, true negative and false positive [10].

b. Precision:

Precision is the ratio between true positive and all positive (True positive and False positive). It is also called Positive Predicted Value. The term of precision:

$$\text{Precision} = (\text{TP} / \text{TP}) + \text{FP}$$

c. Recall:

Without positive classes we have predicted correctly, how many are actually positive. Recall is actually measurement of identifying true positives. The term of recall:

$$\text{Recall} = (\text{TP} / \text{TP}) + \text{FN}$$

d. F1-score:

We cannot compare two models easily so we use f1-score to make them comparable. It is the harmonic mean of recall and precision.

$$\text{F1-score} = (2 * \text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

IV. RESULT ANALYSIS

The frame-work of our research started with a dataset of 3000. While working with that dataset, Random Forest provided a higher value of accuracy than others. The provided accuracy was 62%. When the data was extended to 6000, Random Forest again held the highest value of 68.4% with an increase of 6.4% in accuracy. For not being as our expectation we had to increase the data to 7500. After doing so, at the third time it showed a quite satisfying result which we can afford. It was very much noticeable that in every algorithm, it had an average increase of 5% accuracy. For acquiring the best result cross validation was executed for each and every single algorithm. But the acquisition was a little bit disappointing. With the help of the figure 6 comparison of test accuracy and cross validation score was made. For Naive Bayes, Random Forest, Decision Tree Classifier, SVC, KNN and XGB Classifier the presented test accuracy score in TABLE I was 62.49%, 75.38%, 63.2%, 72.87%, 65.67% and 73% as well. But the Cross Validation score shown in the table was 58.95%, 73.925%, 63.25%, 71.77%, 63.2% and 70.85% respectively. For NB, RF, SVC, KNN and XGB Classifier there was a significant decrease of 3.54%, 1.46%, 1.1%, 2.47% and 2.15% respectively. But for the Decision Tree Classifier there a little increase of 0.05% was seen.

Despite a thorough decrease of accuracy after Cross Validation it was proved that Random Forest had the best scenery of score with 75.35% value of perfection.

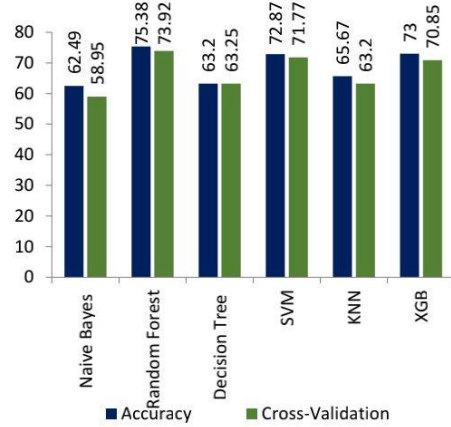


Fig. 6 Comparison between test and cross validation accuracy of different machine learning technique

The precision, recall, F1-score and support were measured to analyze every classifier's performance. In the comparison, variation of the value of each other's parameters was noticed. By the table 1, it has appeared as the comparison of the utilized classes. In the case of Naive Bayes, for the interrogative or class 0 precision, recall and f1-score propped up a very near score each other where the support was 358. But for exclamatory or class 1 it was noticed that precision was less than the class 0 with recall and f1-score was acceptable with 405 support. On the other hand, for assertive or class 2 in case of naive bayes, every parameter was very poor except support.

TABLE I. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (NAIVE BAYES ALGORITHM)

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
62.49	58.95	Interrogative	0.67	0.69	0.68
		Exclamatory	0.61	0.69	0.65
		Assertive	0.58	0.48	0.53

In the case of table 2, For Interrogative or class 0, precision, recall, f1-score were found as 0.79 and support was found as 358. For class 1, precision and f1-score were decreased to 0.72 and .77 respectively. Moreover, recall increased remarkably to 0.82 with support 405. For Class 2, except precision all the parameters decreased.

TABLE II. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (RANDOM FOREST ALGORITHM)

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
75.38	73.92	Interrogative	0.79	0.79	0.79
		Exclamatory	0.72	0.82	0.77
		Assertive	0.75	0.64	0.69

For the Decision Tree algorithm, table 3 presented that precision, recall, f1-score and support were traced as 0.65, 0.68, 0.66 and 488 respectively for class 0. For class 1, the value of precision, recall and f1-score were traced as 0.64 with 538 support. For class 2, precision, recall, f1-score and support were perceived as 0.60, 0.58, 0.59 and 478 respectively. On the other hand, table4 accomplished that Test accuracy has increased a little bit where the value extends to 63.25. But the increase was negligible.

TABLE III. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (DECISION TREE ALGORITHM)

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
63.2	63.25	Interrogative	0.65	0.68	0.66
		Exclamatory	0.64	0.64	0.64
		Assertive	0.60	0.58	0.59

In table 4, all the parameters of the support vector machine are placed. It represented precision as 0.74, recall as 0.74, and f1-score as 0.74 and support as 488 for class 0. Additionally, precision, recall, f1-score and support for class 1 were observed as 0.71, 0.79, 0.75 and 538 respectively. For class 2, precision as 0.76, recall as 0.67, f1-score as 0.71 and support as 474 were obtained.

TABLE IV. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (SUPPORT VECTOR MACHINE ALGORITHM)

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
72.87	71.77	Interrogative	0.74	0.74	0.74
		Exclamatory	0.71	0.79	0.75
		Assertive	0.76	0.67	0.71

In table 5, for the K-Nearest Neighbors algorithm, it was perceived that precision was as 0.59, recall as 0.79, f1-score as 0.67 and support as 483 for class 0. Additionally, for class 1, precision, recall, f1-score and support were traced as 0.69, 0.68, 0.68 and 519 respectively. On the other hand, precision as 0.75, recall as 0.50, f1-score as 0.60 and support as 498 for class 2 was shown in table 5.

TABLE V. K-NEAREST NEIGHBOR ALGORITHM PRECISION, RECALL AND F1-SCORE

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
65.67	63.2	Interrogative	0.59	0.79	0.67
		Exclamatory	0.69	0.68	0.68
		Assertive	0.75	0.50	0.60

For The XGBoost classifiers, table 6 has represented the value of precision as 0.74, recall as 0.77, and f1-score as 0.67 and support as 483 for class 0. On the other hand, precision, recall, f1-score and support were shown as 0.69, 0.68, 0.68 and 519 respectively for class 1. Moreover, for class 2, precision was traced as 0.74, recall as 0.60, f1-score as 0.70 and support as 498.

TABLE VI. MEASURE PRECISION, RECALL AND F1-SCORE FOR THREE CLASSES (XGBOOST ALGORITHM)

Test Score	Cross Validation Score	Class	Precision	Recall	F1-Score
73	70.85	Interrogative	0.74	0.77	0.75
		Exclamatory	0.71	0.77	0.74
		Assertive	0.74	0.66	0.70

To represent the comparison clearly, the graph was drawn. Fig. 7 was charted for class 0. In that chart, Random Forest was the highest scored model for precision, recall and f1-score. Thus, KNN model was also present in that graph containing a high recall score of 0.79. Chart of class 1 was illustrated in Fig. 8. Among six classifiers, Random Forest again gained the top most placed for three parameters. Additionally, Fig. 9 was charted for Class 2, which conveyed that SVC was the top model with high values of the parameters. Moreover, Random forest also gained a very near score of SVC. Ending after all comparison, it is dispensed that Random Forest, SVC and XGB classifiers have delivered much better results than Naive Bayes, Decision Tree and KNN. Yet, Random Forest has delivered the winning score in most comparisons.

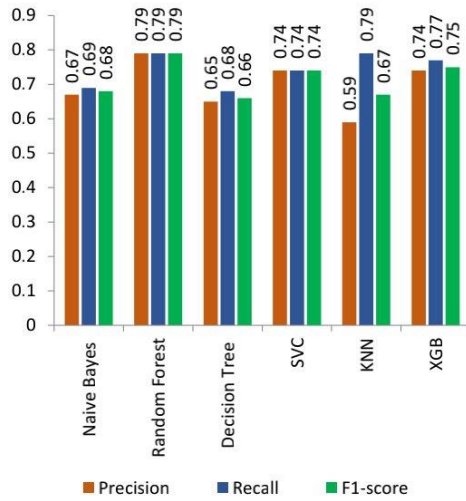


Fig. 7 Comparison among precision, recall and f1-score for class 0 (Interrogative Sentence)

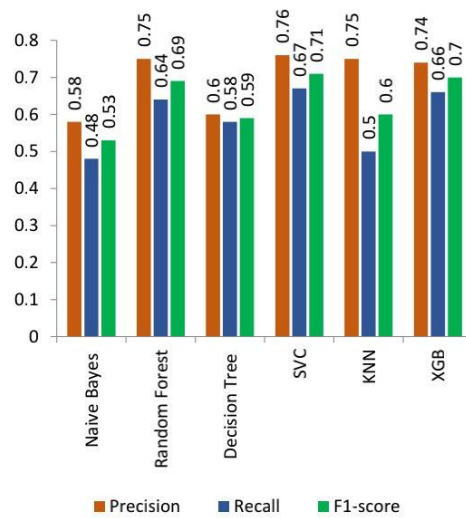


Fig. 9 Comparison among precision, recall and f1-score for class 2 (Assertive Sentence)

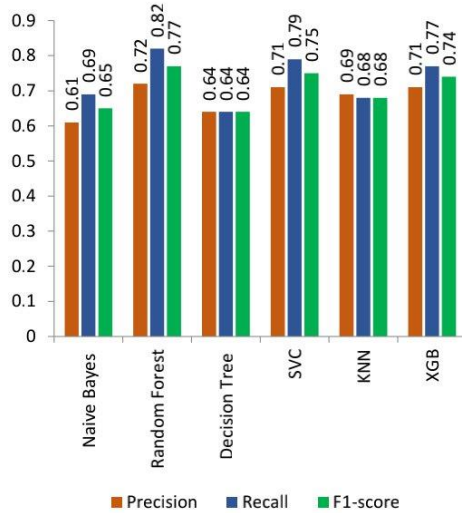


Fig. 8 Comparison among precision, recall and f1-score for class 1 (Exclamatory Sentence)

V. CONCLUSION

The paper proposes the comparison among the classifiers of machine learning and finds out the best approach while the classification of Bengali sentences. In this study, Random Forest outperformed for classifying the different kinds of Bengali sentences with 75.38% accuracy. Hence, if the dataset can be increased to a large margin the outcome would be more productive. Hereafter, Deep learning algorithms will be executed on this dataset by appending the remaining functional sentences as classes.

REFERENCES

- [1] M. Z. Hasan, S. Hossain, M. Arif, and M. Shohel, "Content based Document Classification using Soft Cosine Measure," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 4, 2019.
- [2] T. Nomponkrang, the Department of Computer Education, Faculty of Technical Education, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, C. Sanrach, and the Department of Computer Education, Faculty of Technical Education, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, "The comparison of algorithms for Thai-sentence classification," *Int. J. Inf. Educ. Technol.*, vol. 6, no. 10, pp. 801-808, 2016.
- [3] C. Gu, M. Wu, and C. Zhang, "Chinese sentence classification based on convolutional neural network," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 261, p. 012008, 2017.
- [4] C. Liu and X. Wang, "Quality-related English text classification based on recurrent neural network," *J. Vis. Commun. Image Represent.*, vol. 71, no. 102724, p. 102724, 2020.
- [5] M. Adnan Khan, R. Ali Naqvi, N. Malik, S. Saqib, T. Alyas, and D. Hussain, "Roman Urdu news headline classification empowered with machine learning," *Comput. mater. contin.*, vol. 65, no. 2, pp. 1221-1236, 2020.
- [6] F. Peng and X. Huang, "Machine learning for Asian language text classification," *J. Doc.*, vol. 63, no. 3, pp. 378-397, 2007.

- [7] P. Zhang and Y. Fang, "Research on text classification algorithm based on machine learning," *J. Phys. Conf. Ser.*, vol. 1624, p. 042010, 2020.
- [8] Pal, M. (2005). Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 26(1), 217–222.
- [9] Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3), 660–674.
- [10] Forman, G. An extensive empirical study of feature selection metrics for text classification. *The Journal of Machine Learning Research*. 2003; 3: 1289–305.
- [11] McCallum, A. , Nigam, K. A comparison of event models for naive bayes text classification. In: *Proc of the AAAI-98 Workshop on Learning for Text Categorization*, Citeseer; 1998: 41–8.
- [12] Vieira, J. P. A., & Moura, R. S. (2017). An analysis of convolutional neural networks for sentence classification. *2017 XLIII Latin American Computer Conference (CLEI)*, 1–5. IEEE.
- [13] Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 45–66.
- [14] Han, E.-H., Karypis, G., & Kumar, V. (2001). Text categorization using weight adjusted k-nearest neighbor classification. In *Advances in Knowledge Discovery and Data Mining* (pp. 53–65). Berlin, Heidelberg: Springer Berlin Heidelberg.
- [15] Chen, J., Huang, H., Tian, S., & Qu, Y. (2009). Feature selection for text classification with Naive Bayes. *Expert Systems with Applications*, 36(3), 5432–5435.
- [16] Sentence Classification Dataset [online]. Available: <https://drive.google.com/file/d/1arhxhM5LfcGDJNYixbsMs508YZMn7CPF/view?usp=sharing> [Accessed: 09-Jun-2021]

PLAGIARISM REPORT

Report_Final

ORIGINALITY REPORT

30% SIMILARITY INDEX	14% INTERNET SOURCES	26% PUBLICATIONS	7% STUDENT PAPERS
--------------------------------	--------------------------------	----------------------------	-----------------------------

PRIMARY SOURCES

1	Antara Biswas, Musfiqur Rahman, Zahura Jebin Orin, Zahid Hasan. "Bengali Functional Sentence Classification through Machine Learning Approach", 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021 Publication	19%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	dokumen.pub Internet Source	1%
4	www.marketwatch.com Internet Source	<1%
5	Submitted to Hellenic Open University Student Paper	<1%
6	K. Rithesh. "Chapter 3 Anomaly-Based NIDS Using Artificial Neural Networks Optimised with Cuckoo Search Optimizer", Springer Science and Business Media LLC, 2019 Publication	<1%