Sentiment Analysis in Bengali Text using NLP

by

Ankon Sarkar
18301273
Aishwarja Paul Sourav
18301078
Rezvi Ahmed
18301226

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Ankon Sarkar

Ankon Sarkar

18301273

Sourav Paul

Aishwarja Paul Sourav

18301078

Rezvi Ahmed

Rezvi Ahmed

18301226

Approval

The thesis titled "Sentiment Analysis in Bengali Text using NLP" submitted by

1. Ankon Sarkar (18301273)
2. Aishwarja Paul Sourav (18301078)
3. Rezvi Ahmed (18301226)

Of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on Jan 19, 2023.

Examining Committee:

Supervisor:

Mr. Arif Shakil

Lecturer
Department of Computer Science and Engineering
BRAC University

Co-Supervisor:

Farig Yousuf Sadeque

Dr. Farig Yousuf Sadeque

Assistant Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Natural Language Processing, a branch of AI, teaches computers to understand speech and text in multiple languages. Machine learning or deep learning techniques can be used to develop rule-based models of human-spoken languages to simulate accurate text-meaning predictions. Although many studies have vastly improved the categorization of text data in languages such as English, Arabic, Chinese, Urdu, Hindi, etc, Bengali text categorization has not progressed much compared to others. This research proposes an approach to analyzing and extracting basic emotions (Happiness, Sadness, Fear, Anger, Disgust Surprise) from Bengali text data. This can be done by gathering real-life data and producing a special rule-based algorithm using supervised machine learning and deep learning techniques. We evaluate the performance of our models using our own dataset BANEmo, consisting of 14999 annotated Bengali text data. To make text data machine-readable, we employed Bag of words, TF-IDF, Glove, and BERT embedding. We measured performance using supervised machine learning models like Naive Bayes and Support Vector Machine. Deep learning techniques like LSTM and Transformers (BERT) were also implemented. Our BERT model outperformed others with an overall accuracy of 69.2%.

Keywords: Natural Language Processing, Sentiment Analysis, Bangla Text, Machine Learning, Deep Learning, LSTM, Transformers, BERT.

Dedication

The paper is written in honor of our families and co-workers. The team members' perseverance and the family's unwavering support were crucial in the achievement of this paper. Without the assistance of our esteemed supervisor and co-supervisor, who has been a constant source of guidance and advice, we could not have completed our thesis. This paper is also dedicated to them.

Acknowledgement

We are eternally grateful to the Almighty, without whom we could not have completed our thesis. Words cannot explain how thankful we are to our supervisor, Mr. Arif Shakil Sir, for his time and suggestions. Also, without the kind assistance, knowledge, time and skill of our honorable Dr. Farig Yousuf Sadeque Sir, our work would not have been completed. We would like to thank them for their helpful feedback and moral support in our effort. Finally, a special thanks to our amazing parents, whose prayers and support enabled us to finish our thesis work.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$BERT$ Bidirectional Encoder Representations from Transformers

BoW Bag of words

$GloVe$ Global vectors for word representation

GRU The Gated Recurrent Unit

$LSTM$ Long Short Term Memory

RNN Recurrent Neural Network

SA Sentiment Analysis

SVM Support Vector Machine

$TF-IDF$ Term Frequency-Inverse Document Frequency

Chapter 1

Introduction

Sentiment analysis is the process of predicting the sentiment or emotions of a set of text data. It is a method where text data from social media websites, blogging sites, books, reviews from eCommerce websites, and online portals can be taken and processed to converge towards a set of sentiments that are spread throughout the majority of that block of text/data. Because of the ease of data gathering and relevance to NLP, sentiment analysis has garnered the interest of a vast pool of researchers across the world. Emotion Detection is the process of identifying human emotion from any written text by analyzing it. One of the core applications of NLP, Research in this field is crucial for enhancing human-machine communication. Bengali is one of the most spoken languages in the world. Thus, being able to extract and categorize emotions from Bengali texts can benefit many individuals. By articulating shared linguistic traits, it will also advance SA research.

Moreover, In current times people are sharing their thoughts on different topics thanks to social media. This has increased the number of public opinions that are circulating each day around social media. These opinions can be accessed by anyone, which may cause a disruption in the harmony of our society. But if we know the tone of an article before we get into the details, we can figure out if it's right for the audience it might reach and, if so, what effects it might have. With the help of real-life data and Sentiment analysis, we can solve this problem. In this paper, we have developed models using supervised machine learning and deep learning techniques like Naive Bayes, Support Vector Machine, Recurrent Neural Networks & BERT. We have created our own manually annotated Bengali text dataset named "BANEmo" consisting of 14999 text data collected from the comments sections of Facebook, Youtube, and some renowned online Bengali news portals like Prothom Alo, BBC Bangla, DW Bangla, etc. The dataset was classified in six basic human emotions [1] along with "sarcasm" & "others" classes. We have applied various kinds of word embedding techniques like Bag of Words, TF-IDF, GloVe, etc. to figure out correlated words and predict the emotions portrayed by them. Our developed models will classify the Bengali text data to distinguish and extract basic human emotions.

1.1 Research Problem

In current times, with the free flow of data, people are constantly flooded with an overflow of information whenever they log into their social media accounts or tune into any online portal that may contain news or articles about various topics. In this situation, people tend to focus on an article according to their attention span, which has come down to only 8.25 seconds in 2022. Within this short period of time, people cannot correctly grasp the core gist of an article, let alone comprehend the entirety of it. This leads to the misinterpretation of information and the spread of misinformation consequently. Misinformation can lead to many harmful things such as disruption of harmony among people, wars between governments, Degradation of interpersonal relationships etc.

An extent of this problem can be the intentional incivility and spread of hate speech in the mass communication mediums in the forms of cyberbullying and cyber harassment [17]. This ultimately leads to mental health issues and dissociation from social media among social media users [18].

It also results in a substratum of internet users quitting the regular usage of the internet and aids made available by it, such as online health forums. Which might cause elongated periods of mental depression and disdain for the victims of cyber abuse [4].

These problems can be solved by classifying blocks of texts according to their expressed emotions with the help of Sentiment Analysis through building predictive models. We can distinguish between the polarity of sentences and identify hate speech [22] by associating a score to them that can be calculated by developing an algorithm using supervised machine learning where we can feed the model prelabeled data to teach it how to detect patterns and distinguish between different basic emotions [14].

The core reason behind the development of the world wide web was to make information about anything more accessible, and it has done just that. However, with this advancement came a few inevitable demerits that we need to think about in order to enhance the philosophy behind the internet. As there is no way to classify all data before publication, people are gaining access to information that is not appropriate for them. For example, children are gaining access to vulgar content that may prove to be harmful to the healthy development of their mental growth. Besides, people of different cultures are gaining access to information about each other that is in no way relevant to them which is encouraging the trend of being strongly opinionated towards everything and everyone they come across. We can observe a good example of this in our localities, where people are ruining their own harmony over warfare and political problems happening in countries that are distantly relevant to them [21].

This problem can be addressed by processing text data for locating biases and comparative statements to figure out the primary objective of a block of text. This way, we can clearly distinguish between the subject that is being demeaned, defamed, or criticized, which will ultimately help us to calculate the impact of that set of statements among the audience and moderate its monetization of it accordingly.

Finally, as articles and stories these days contain a variety of components compared to those of ancient times, it has become harder for people to categorize and classify them into groups. This can lead to the misrepresentation of information and obstruct authors from achieving their desired reach.

We can resolve this issue with Sentiment Analysis by breaking these components down and taking them into account individually [24]. Such as, we can train the machine learning model with similar data that contains emojis, idioms, sarcasm, and similar component that are more concurrent in the current articles to teach the model a pattern so that it can detect the context of a sentence and predict a possible meaning of it to provide it with a score that depicts its genre and nature.

Our research addresses all of the issues mentioned above and works on creating a framework that can possibly solve each of the problems effectively.

1.2 Research Objectives

Due to the advancement of technology and the rapid increase of internet users, a huge amount of data is generated on a regular basis. Sentiment analysis has generated much interest in natural language processing due to the growth of digital social material on the Internet. The data which is generated is mostly textual data and unstructured in nature. An NLP-based automated text classifier assists in automatically categorizing texts into pre-defined groups. The availability of vast quantities of internet data and the development of machine learning algorithms have facilitated the creation of a variety of approaches for evaluating sentiment and emotions from writings in Arabic, English, French, and many other languages [13]. At the present time, the Bengali language is used in social media, news, articles, product review, and in many other fields. So, analyzing sentiments in the Bengali language is an interesting topic to study. The objectives of our research are:

- To understand how sentiment analysis works in the Bengali language by using Natural Language Processing tools.
- To create a framework that analyzes sentiment and extracts emotions from Bangla text data.
- To distinguish between at least six sentiments/emotions found in the texts and classify them successfully. For example; happiness, sadness, disgust, anger, fear, surprise, sarcasm and undefined.
- Represent performance-based analysis of Bangla text categorization using machine learning techniques.

Lastly, we want to evaluate the model so that, based on the evaluation, we can point out limitations and offer further recommendations for future improvements and references.

Chapter 2

Related Work

2.1 Literature Review

2.1.1 Sentiment Analysis

Sentiment analysis, often known as opinion mining, is an NLP method for evaluating people's emotional reactions to a scenario or event. An individual's assessment or evaluation of a topic or event determines whether their opinion is positive, negative, or neutral on a particular topic or matter. In addition to polarity, it can also detect certain feelings or emotions such as; anger, happiness, sadness, Etc. Machine learning frameworks have gained substantial attention in determining the sentiment of different kinds of data utilizing sentiment as a polarity, such as positive, negative, or neutral [23]. Today, Sentiment analysis is increasingly gaining traction as a critical technique for monitoring and interpreting sentiment in all types of data. Next, the Classification of sentiment is achievable at various levels, including the Sentence, Aspect and Document levels, among others. When assessing the document's positivity or negativity, the document itself is the primary unit of analysis. Sentence-level sentiment categorization, on the other hand, splits each sentence into two groups: subjective and objective, followed by positive, negative, and neutral. Lastly, finding and extracting product attributes from the source data is the focus of aspect or feature-level sentiment analysis [9].

The possibilities of Sentimental Analysis are endless and can be applied to various modern-day industries. Some of the popular and common applications of SA could be product analysis, customer feedback analysis, chatbot, media monitoring etc.

2.1.2 Natural Language Processing (NLP)

Natural Language Processing (NLP) is a set of computing algorithms that are theoretically driven for the automatic interpretation and analysis of text data [2]. This is a subset of artificial intelligence that reads, understands, and interprets human languages. Applications of NLP research include sentiment analysis, text summarizing, topic modeling, information extraction, question answering, and machine translation. In addition, since there is a large quantity of opinionated data on the Internet, sentimental analysis is becoming an important topic in NLP and machine learning. This field is progressively drawing the focus of researchers.

Language Processing Pipelines are used by NLP to read, decode, and interpret human languages.

2.1.3 Related works

Deep learning-based models were suggested by Tripto et al. in [13] for classifying Bengali sentences with a three-class (positive, negative, neutral) and a five-class (strongly positive, positive, neutral, negative, and highly negative) sentiment label. They also developed a model which identifies and categorizes the emotions in a Bengali sentence, such as happiness, surprise, anger, sadness, disgust, and fear. They gathered their datasets (YouTube comments) from the most popular Bengali videos. For opinion mining, they employed LSTM with a deep network and CNN as its main layer. In addition, they utilized SVM and Naive Bayes to identify sentiments and emotions.

Authors Hassan et al. [7] provided pre-processed substantial textual datasets of both Bengali and Romanized Bengali texts and concentrated exclusively on deep learning techniques. They contributed to a readily usable pre-processed dataset of 10,000 Bengali and Romanized Bengali annotated by adult native Bengali speakers. In addition, they worked on applying deep recurrent models to the Bengali and Bengali-Romanized textual samples. In addition, they examined the dataset in Deep Recurrent Models, especially the LSTM, utilizing two types of loss functions: binary cross-entropy and categorical cross-entropy.

Xu et al. [20] shows that the main factor in determining the sentiment pattern in textual sentiment analysis is the keywords that express human sentiment. As a result, creating an efficient and detailed sentiment dictionary for sentiment analysis is quite important; otherwise, it may ignore many sentimental words during the analysis. In this proposed method a weight should be assigned to sentimental words, negative words, adverbs and conjunction, and all types of words. Also, the number of adverbs and negative words are used to calculate the sentiment values of the sentences. In addition, an extensive sentiment dictionary includes fundamental, field, and polysemic sentiment words. Finally, to classify the field of the textual space where the polysemic emotional word appears and to identify the sentiment score of the word, a Naive Bayesian field classifier is being implemented.

Ashik et al. [15] developed a genuine and reliable dataset for sentiment analysis of the comments of the readers of a Bengali newspaper in the comment box. A textual unit's emotion can be categorized in one of three ways; ML, Lexicon-based, or Hybrid approaches. When the data has exactly two classes, they propose to use the SVM classifier model. To classify data, it finds the best possible hyperplane that separates each set of points into its own category. Next, they used RNN and LSTM to create a memory for the model in order to deal with the proper context of the words used. In addition, CNN, a type of deep neural network, was employed to address the issue of data overfitting.

Sadeque et al. [17] used the RNN model that was trained on a database of newspaper commentary to detect incivility in online newspaper comment sections as well as on Twitter. Incivility detection plays an important role in recognizing disrespectful language, which is an important part of sentiment analysis. They categorized two major types of incivility, name-calling, and vulgarity. They mainly used machine learning models to detect them.

Sourav et al. [26] propose a collection of methods for identifying and extracting Bangla emotions. Using transformer-based models, they classify Bangla words as anger/disgust, fear/surprise, joy, and sadness. They evaluated their models using the dataset "UBMEC". This dataset combines two publicly accessible Bangla corpora on 6 key emotion classes, BNEmo and BEmoC. Data preprocessing included Stopwords and duplicate drops. Their model was m-BERT-trained. m-BERT outperformed ML techniques, achieving 69.0% accuracy for four classes.

Rahman et al. [16] compiled a text corpus of user comments from several Facebook groups discussing socioeconomic and political issues and attempted to extract the six core emotions conveyed therein. Finally, they analyzed the performance of five popular classical machine learning algorithms employing different sets of features: Naive Bayes, Decision Tree, k-NN, SVM, and K-Means Clustering. F1 score (macro) of 0.3324 was attained by their top model (SVM with a non-linear RBF function kernel), with an average accuracy of 52.98%.

From the above discussion, it is observed that most of the researchers use Deep Learning, Supervised Machine Learning, Neural Network Model, and Lexicon based Dictionary based approach to analyze sentiments and extract emotions from text data. For example, Supervised ML methods like SVM, Naive Bayes, and Neural Networks and Deep learning methods like LSTM, and CNN are applied to classify and fit the text data. However, there are some unique challenges one might face while working on SA in the Bengali language. Because proper datasets and resources on the Bengali language are rare. Moreover, the grammatical structure of the Bengali language is quite complex. As a result, applying proper and suitable algorithms and models could be difficult.

Chapter 3

Dataset and Preprocessing

3.1 Data collection

The accurate conclusion of a study depends on the data collected. It is challenging to conduct a significant study if data collection and implementation are not perfect. With the goal of creating a novel dataset, we culled Bengali text data from multiple web platforms, including social media, to build our "BANEmo" dataset. There are 14,999 different pieces of Bengali text included in the "BANEmo" Bengali emotional analysis dataset. When compared to other publicly available Bengali SA datasets, this one is far larger. Information was gathered from numerous social media platforms, including but not limited to Facebook, YouTube, and others. Besides, a lot of information was gathered from the comment section of popular online news portals like Prothom Alo, BBC Bangla, DW Bangla, Prothom Alo, Samakal, etc. While collecting Bengali text data for SA we faced some difficulties. Different sources had different types of Bengali text that represented different emotions. And the tricky part was that the text data were unbalanced when it comes to expressing emotions. For example, some particular posts or news portals contained Bengali comments that expressed happiness and others expressed sadness, anger, etc. It is rare to find all the emotions expressed equally in a particular post. Again, many text data contained spelling and grammatical mistakes. Dealing with these difficulties was quite challenging.

3.2 Dataset description

We build our own corpus BANEmo, which contains all Bengali comments for sentiment classification. In the 1970s, Paul Eckman, a psychologist, defined six basic emotions that he believed were shared throughout all human societies. Happiness, sadness, disgust, fear, surprise, and anger were the basic feelings that he identified [1]. We have labeled our dataset as these six emotions. Furthermore, we added sarcasm and undefined categories for those texts that didn't fall under those basic emotions. So in total, we labeled our dataset as follows:

- **Happiness:** A positive mental and emotional state characterized by happiness, fulfillment, and contentment.
- **Sadness:** A mental or emotional condition characterized by sadness, despair, or a lack of hope.

- **Anger:** A negative mental or emotional state characterized by anger and resentment.
- **Disgust:** An intense feeling that causes repulsion.
- **Fear:** A basic human reaction that can mean the difference between life and death.
- **Surprise:** A brief emotional state, either positive or negative, following something unexpected.
- **Others/Undefined:** Any text that does not fall under the aforementioned definitions.

3.3 Data labeling

The collected data were manually classified into eight distinct categories: happiness, sadness, disgust, fear, surprise, anger, sarcasm, and the undefined. A total of three native Bengali speakers manually annotated the data. For the sake of accuracy, we had each person annotate the text data independently of one another. For the final labeling, emotions with the most votes were selected for each text data.

Example of dataset annotation:

Text data	1st annotator	2nd annotator	3rd annotator	Final labeling
বিশ্ববিদ্যালয়ের তালিকায় বাংলাদেশের বিশ্ববিদ্যালয় আছে!!!!!!	Surprise	Surprise	Surprise	Surprise
নোংরা রাজনীতি যতদিন বিশ্ববিদ্যালয়ে থাকবে ততদিন ভালো কিছু আশা করা যায় না।	Disgust	Disgust	Disgust	Disgust
যতটুকু দিলে সাধারণ মানুষের জীবন অতিষ্ঠ হয়, ঠিক ততটুকু।	Disgust	Fear	Disgust	Disgust
পুলিশের উচিত ছিলো মানুষের সেন্টিমেন্ট বুঝে কথা বলা	Anger	Anger	Disgust	Anger
দেশের টাকা গুলো এভাবেই চলে যাচ্ছে	Sadness	Fear	Sadness	Sadness

Table 3.1: Final annotation based on majority vote

In case, no emotion got the majority vote, for example three annotators labeled the text data as three different emotions, we asked an meta annotator who has good knowledge in Bengali to help us annotate the text data and break the tie.

For example:

Text data	1st annotator	2nd annotator	3rd annotator	Final labeling
আমাদের দেশে বিশ্ববিদ্যালয়গুলোতে তো আর লেখাপড়া হয়না। শেখানো হয় রাজনীতি	Sadness	Fear	Disgust	Sadness
এই আনন্দের দিনে যারা মুখ গোমড়া করে থাকবে তাদের বিরুদ্ধে মামলা দায়ের করা যেতে পারে।	Disgust	Sadness	Sarcasm	Disgust

Table 3.2: Final Annotation Done by Meta Annotator

Finally, we have considered final labeling as our key sentiment for this dataset.

The total 14999 collected text data was annotated in the following distributions:

Emotions	Total
Happiness	4130
Sadness	4180
Fear	787
Anger	1752
Disgust	3441
Surprise	352
Sarcasm	155
Undefined	202

Table 3.3: Number of comments in each class

3.4 Data pre-processing

Remove duplicate and null values: There were a total of 95 duplicate comments and 1 null value. We have dropped those duplicate and null values. After dropping those comments we have 14903 text data remaining.

Text cleaning: The goal of NLP (Natural Language Processing) is to train computers to understand natural language, and text cleaning is the process by which raw text is prepared for NLP. The Data Cleaning procedure filters comment language by removing irrelevant elements. The major collection of comments was cluttered with emojis, punctuation, and other unwanted elements.

- **Punctuation removal:** We stripped all punctuation from the comment text data to improve categorization accuracy. Their presence can be equated to background noise.
- **Emoticons removal:** Emoticons, hashtags, and other unwanted things were removed to make the data noise-free and efficient. Removing emoticons also enabled the annotators to label the text data neutrally.

Before Cleaning	After Cleaning
বিশ্ববিদ্যালয়ে শিক্ষক রাজনীতি, ছাত্র রাজনীতি চিরতরে নিষিদ্ধ না করলে শিক্ষার মান যেটুকু আছে তাও হারিয়ে যাবে।	বিশ্ববিদ্যালয়ে শিক্ষক রাজনীতি ছাত্র রাজনীতি চিরতরে নিষিদ্ধ না করলে শিক্ষার মান যেটুকু আছে তাও হারিয়ে যাবে
শিক্ষাই আলো,,অনেক দিন ধরে জ্বলছে তো, তেল ফুরিয়ে গেছে। ।	শিক্ষাই আলো অনেক দিন ধরে জ্বলছে তো তেল ফুরিয়ে গেছে
এরা শিক্ষিত পাগল��	এরা শিক্ষিত পাগল
গাধা পানি খায় তবে ঘোলা করে খায় □□□□□□	গাধা পানি খায় তবে ঘোলা করে খায়
এখন অনেক শিক্ষকই (সবাই না) রাজনীতি করে নিজ ভিত্তি প্রস্তর মজমুদ রাখার জন্য। যেটা আগে ছিলো না	এখন অনেক শিক্ষকই সবাই না রাজনীতি করে নিজ ভিত্তি প্রস্তর মজমুদ রাখার জন্য যেটা আগে ছিলো না

Table 3.4: Data cleaning

Remove low length text: Before implementing word embedding we removed low length data. We removed all data which had less than 3 words. Thus we have removed 494 data.

3.5 Exploratory Data Analysis

After performing data preprocessing we get in a total of 14409 text data for our corpus.

Number of data in each class: Sadness sentiment has the most numbers of data instead, sarcasm has the lower number of words in the dataset. Data in each type of sentiment follows:

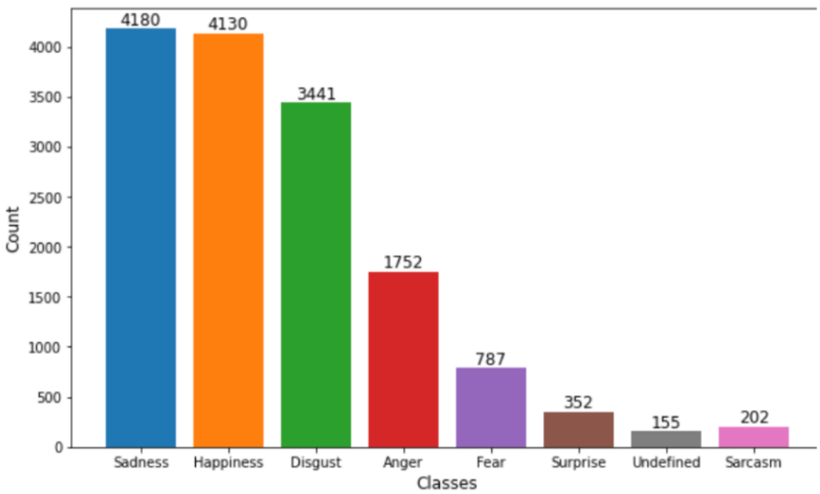

Figure 3.1: Number of data in each class

Data Ratio: We got the following Sentiment ratio in each type of sentiment:

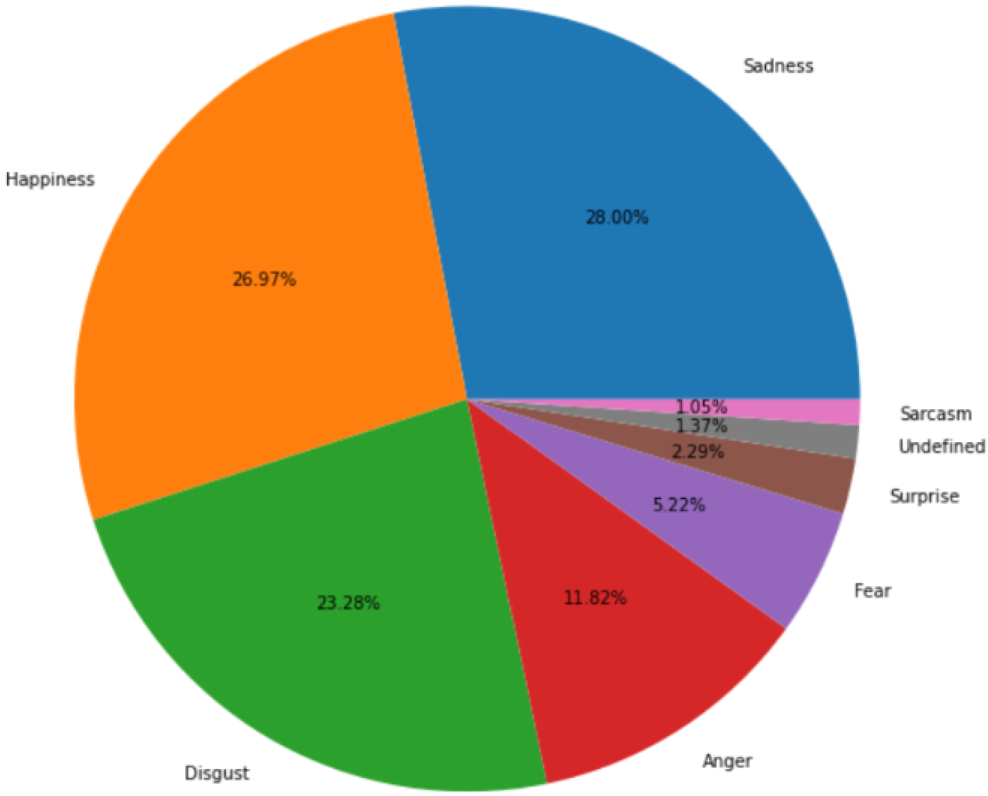

Figure 3.2: Data ratio of each class

Number of Words in each class: Sadness sentiment has the most number of words, on the other hand, sarcasm has the lowest number of words in the dataset. The total number of words in each sentiment is as follows:

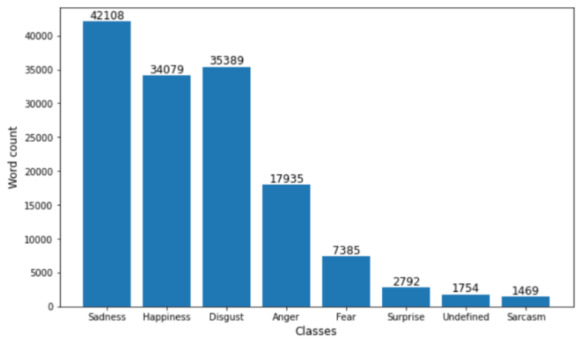

Figure 3.3: Number of Words in each class

Unique Words: We calculated the total unique words in the whole dataset. Total Unique words in Dataset: 19276.

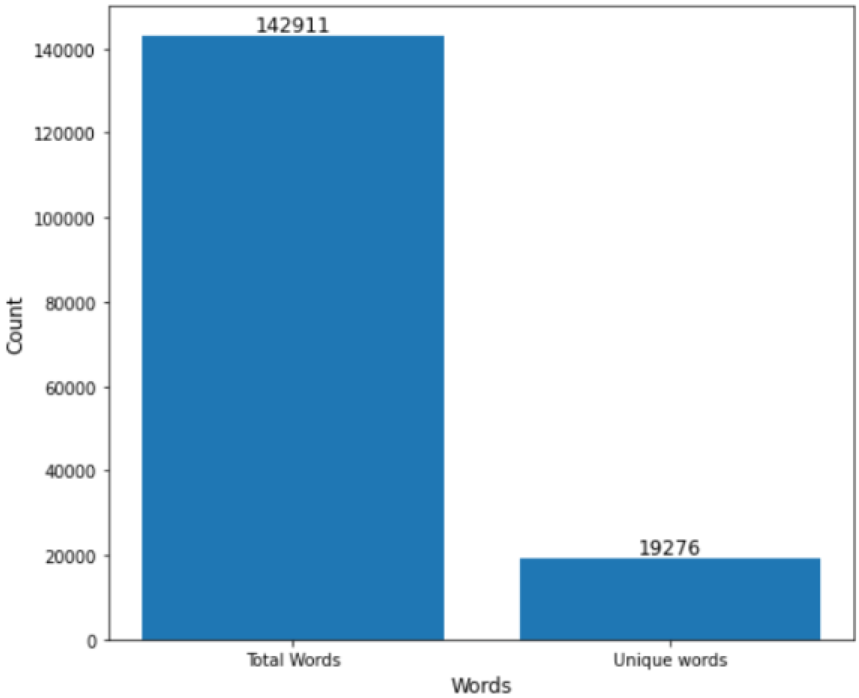

Figure 3.4: Total number of words and Unique words Visualization

Most frequent words: In the corpus, some words were very frequently used. The top 10 frequent words are as follows:

না	2530
এই	1511
করে	1492
আর	1177
জন্য	951
কি	913
হবে	893
ভালো	781
করা	734
মানুষ	729

Table 3.5: Top 10 most frequent words in dataset

Data length distribution: The length of data in the corpus varied a lot. We have found various numbers of words in each data.

Maximum word in a comment	86
Minimum word in a comment	3
Average word in a comment	10

Table 3.6: Data length distribution

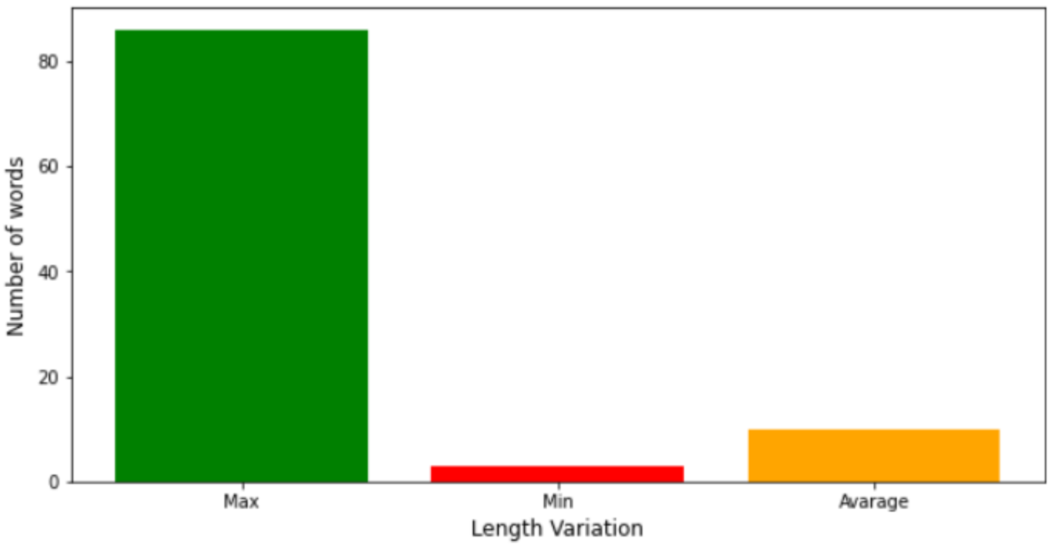

Figure 3.5: Data length Distribution Visualization

Chapter 4

Methodology and Proposed Emotion Detection Model

This section provides an overview of our research methodology. Initially, we collected Bengali text data from the comment sections of some renowned Bengali online news portals, Facebook, and Youtube. To do so, we have extracted Bengali text data by hand to maintain its quality and novelty. Besides, we labeled those text data to categorize them. Next, we applied various data preprocessing techniques to make the dataset more effective such as; removing duplicate comments, dropping null comments, and cleaning the text data by removing emoticons, punctuation marks, and other unnecessary things that might make the data noisy. Besides, we removed some low-length comments to make the dataset more balanced. Next, we represented the text data with Bag of words, TF-IDF, and word embedding techniques. Furthermore, we need to split the dataset into testing data, training data, and validation of data. Then we need to apply baseline machine learning algorithms such as SVM, and Naive Bayes. Then we moved to deep learning and transformers models to classify them into different categories. Next, We need to provide the testing data in the model and evaluate the performance of the selected supervised ML algorithms. Lastly, we will test the performances of different algorithms, and based on the performances, we will be able to offer a better model. Furthermore, we will be able to identify the limitations of that model as well as provide some suggestions for future enhancements and references.

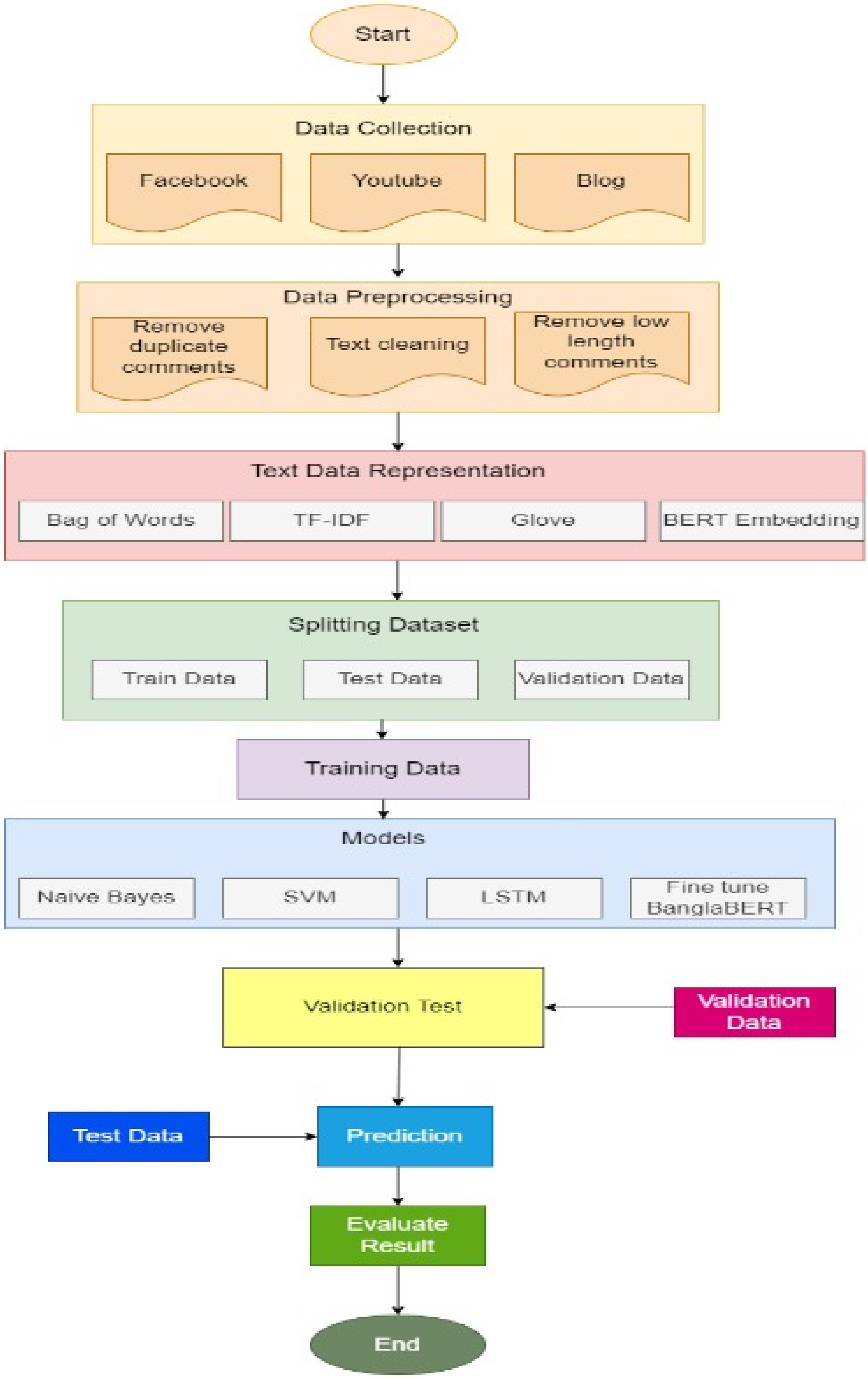

Figure 4.1: Research methodology

4.1 Text representation

Text representation is one of the fundamental tasks to retrieve information from text data. Machines do not understand human language. To represent text data for machine-understandable language, text data needs to be converted into numbers. Text representation is the technique to convert text to number for a machine understandable language.

4.1.1 Bag of words

Bag of words is a text modeling technique of Natural Language Processing. In technical terms, we can say that it is a technique for extracting features from text data. This method provides a simple and flexible method for extracting document features [13]. Using the Bag-of-Words technique, we can transform a text with a variable length into a vector with a fixed length. This model contains a vocabulary of recognized words as well as a measure of the frequency of words in each sentence.

4.1.2 TF-IDF

In information retrieval, the TF–IDF method is used. This gives a number that shows how important a certain word is to the given document dataset. TF-IDF employs weights for text mining and retrieval of information. The value of weight depends directly on how many times a word appears in the given dataset. [12] To facilitate better data retrieval, we have counted the number of times each word appears in the provided document. The search engines take into account the TF-IDF weight included in the query when ranking documents in response to user queries [14]. The frequency with which a term occurs in a text is measured by its "Term Frequency" (TF) value. The significance of a word in a given document is measured by its inverse document frequency (IDF). given a dataset D, a vocabulary w, and a set of documents d ϵ D, typically, we compute:

$$W_d = f(w, d) * log(|D|/f(w, D)$$

Here, D is the dataset, f (w, d) describes how often a phrase appears in a document, and W_d indicates the significance of a term.

4.2 Word Embedding System

4.2.1 Glove Vector

GloVe is a word vector learning approach. GloVe caught the word vector wave. Like words attract and dissimilar words repel in word vectors. GloVe employs word co-occurrence and local context to construct word vectors, unlike Word2vec, a collection of related models.

GloVe is an unsupervised word vector representation method. Training uses corpus-based global word-word co-occurrence statistics, and the generated representations show word vector space's linear substructures.

4.2.2 Model Details

We have implemented two baseline ML models; Naive Bayes and SVM. Besides, we used deep learning models like LSTM, and Transformers (BERT).

4.2.2.1 Naive Bayes

Naive Bayes is a classification method for binary (two-class) and multi-class categories. The method is simplest to read about when described using binary or categorical inputs. It is a binary and multiclass classification algorithm. Naive Bayes works well with categorical input variables compared to numerical ones [9]. It is capable of processing continuous as well as discrete data [14]. It scales well with the number of predictors and data points used. It predicts in real time and operates swiftly. Unneeded features are ignored. The "Naive Bayes" method is a quick classification strategy that calculates the likelihood that a class contains a particular feature quickly by applying the Bayes theorem:

$$\text{P(label/features)} = \text{P(features/label)*P(label)/P(features)}$$

Here, the likelihood that a label will be observed, is denoted by P(label). In the context of feature classification, the prior probability that a set of features is being classed as a label is denoted by P(features|label). The prior probability that a certain set of features has occurred is denoted by the symbol P(features).

After Performing Bag of Words and TF-IDF to represent text data for machine-understandable language. We have performed Multinomial Naive Bayes on both TF-IDF and bag of words.

4.2.2.2 Support Vector Machine(SVM)

SVMs are built on the idea that the best way to differentiate between classes is to locate a linear separator or hyperplane in the search space [15]. Multiple hyperplanes may effectively divide the groups; the one used is the one along which the largest normal distance is found for any of the data points [9]. SVMs, or support vector machines, are one sort of supervised learning model with related learning algorithms for conducting classification and regression analysis on data [14]. The RBF kernel was chosen for this study among the several widely-used SVM kernels.

$$K(x, x') = e^{-}\gamma||x - x'||^2$$

Here, $||x - x'||^2$ is the squared Euclidean distance between two feature vectors (2 points).

In our case, in BoW we have used the SVM RBF function kernel with a gamma value of 0.1 which gives better accuracy than other kernels and gamma values.

In TF-IDF we have also used the SVM RBF function kernel with a gamma value of 0.5 which provides better performance.

4.2.2.3 Recurrent Neural Network

For many NLP tasks, deep learning-based neural network models have proven to be highly effective. RNN, the other prominent neural network architecture, can process sequences of arbitrary length and capture long-term dependencies. LSTM is one of the variants of RNN which stands for Long Short Term Memory.

The fundamental distinction between a standard RNN and an LSTM is that the latter has a persistent memory, while the former does not [5]. We have utilized LSTM to categorize the data into categories in our dataset since we need to capture or process the specific element's entirety. The decision will be a consequence of the entire. Therefore, we need such a mechanism to capture the totality. Long-term dependencies are classified most accurately with LSTM in the majority of cases.

During the training of a deep neural network such as RNN, an unstable behavior known as the vanishing gradient problem may occur. As more layers employing particular activation functions, such as the sigmoid function, are added to neural networks, the gradients of the loss function approach zero, making it difficult to train the network. The LSTM solves the problem of the RNN's vanishing gradient. For concerns with vanishing gradients, RNNs other than LSTM cannot be employed for long-term dependence [19].

LSTMs are made to avoid long-term dependence. They don't struggle to learn; long-term memory is their default tendency. Similar to RNNs, LSTMs have a chain-like structure, but the repeating module is constructed differently. The four neural network layers interact in a unique way [5].

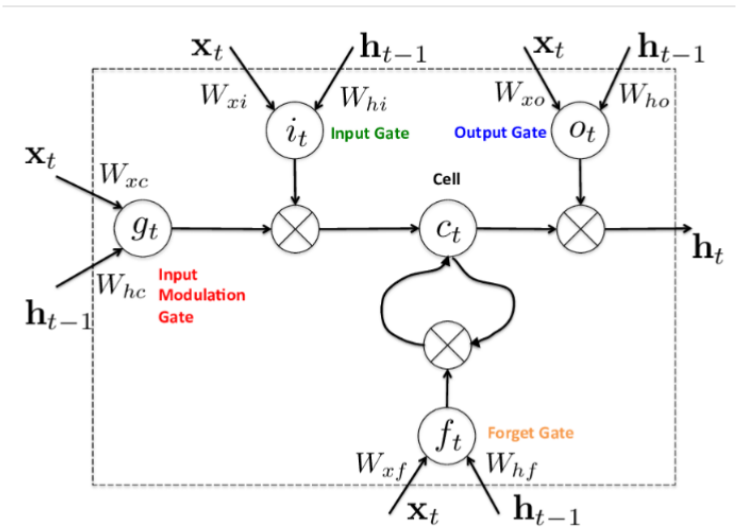

Figure 4.2: LSTM Architechture

The LSTM can remove or add information via gates. Gated information is optional. Pointwise multiplication and sigmoid neural networks are used. The sigmoid layer generates numbers between zero and one to indicate component throughput. Zero means "allow nothing through" and one means "let everything through." Three gates protect and regulate the LSTM cell state.

In our proposed LSTM model there are two Bidirectional layers along with the Embedding layer and Dense layer as illustrated in the figure 4.3

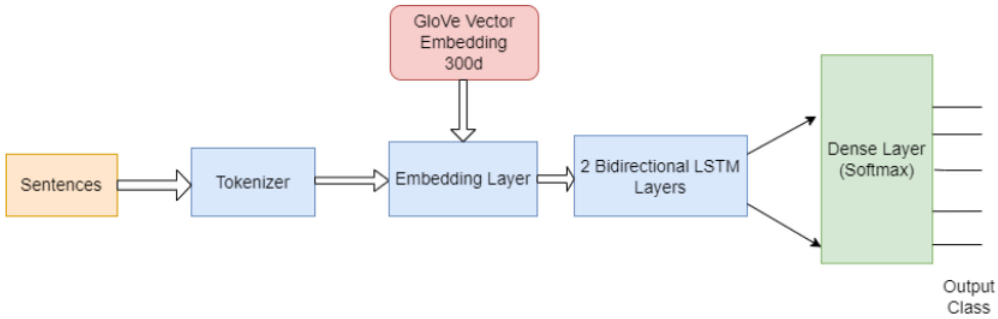

Figure 4.3: LSTM model architecture

4.2.2.4 BERT

Bidirectional Encoder Representations from Transformers (BERT) is an implementation of the Transformers deep learning model, which has each output element connected to each input element and dynamically calculates weightings between them.
Regarding the framework of the BERT model, there are two steps: pre-training and fine-tuning. During pre-training, a large unlabeled corpus is used to train the model. All parameters are fine-tuned using labeled data for the given tasks after the model is initialized with the pre-trained parameters for fine-tuning. Model-wise, BERT is built around a bidirectional, multi-layer Transformer encoder [11]. This encoder has N layers, all of which are the same. There are two sub-layers in each of these layers. The first is a position-wise completely connected feed-forward network, and the second is a multi-head self-attention mechanism. It uses a residual connection at the boundary between the two sublayers, and then normalizes the layers [6], [8]. Each sub-output layer is the norm of LayerNorm(x + Sublayer(x)), where Sublayer(x) is the function implemented by the sub-layer [10].

We utilized a pre-trained BERT model that was trained on a big dataset as a starting point. The process of further training the model using our relatively smaller dataset is known as model fine-tuning. Since the Transformer's self-attention mechanism enables BERT to model numerous downstream tasks or activities, fine-tuning is simple and straightforward. For each task, the particular inputs and outputs are plugged into BERT and all parameters are fine-tuned [11].

In addition, we have implemented the Train the entire architecture fine-tuning method. In this method, we retrain the entire pre-trained model on our dataset and input the results into a softmax layer. In this instance, the error is propagated back through the entire architecture, and the pre-trained weights of the model are adjusted depending on the new dataset.

Chapter 5

Results and Analysis

At the beginning of the sentiment analysis task we randomly split the whole dataset into 3 parts for training and testing purposes. We split into an 80:10:10 ratio to the whole dataset randomly.

Training set: The training data set is fed to the model. The model learns from it and learns any hidden characteristics or patterns. We used 80% data of the whole dataset for training.

Validation set: The validation set is used to evaluate the accuracy of the model throughout the training phase. We can then use the results of the validation to adjust the model's hyperparameters appropriately. The 10% data of the whole dataset was used for validation purposes.

Testing Set: The test set is used after the training compilation, in order to evaluate the model after completing the training. The 10% data of the whole dataset was used for final evaluation.

Number of Training Data	11527
Number of Validation Data	1441
Number of Testing Data	1441

Table 5.1: Number of training, testing and validation data

As we have already discussed, In Our dataset, at first, we classified and labeled our dataset in the six basic emotion classes, such as "sadness", "happiness", "disgust", "surprise", "fear" and "anger". Also, There were some comments which were labeled as "sarcasm" and comments that do not fall into any sort of emotion were marked as "undefined". The number of data in the "surprise", "sarcasm" and "undefined" class were insufficient so we combined them into a new class named "others".

The authors of [3] state that human emotions can be further classified into just five categories: happiness, sadness, fear, and anger/disgust. As, the semantic meaning of a statement from anger and disgust are extremely similar. Therefore, we further divided the dataset into five classes to observe how our model classifies the emotion classes. ”Anger or Disgust” was renamed ”Exasperation”.

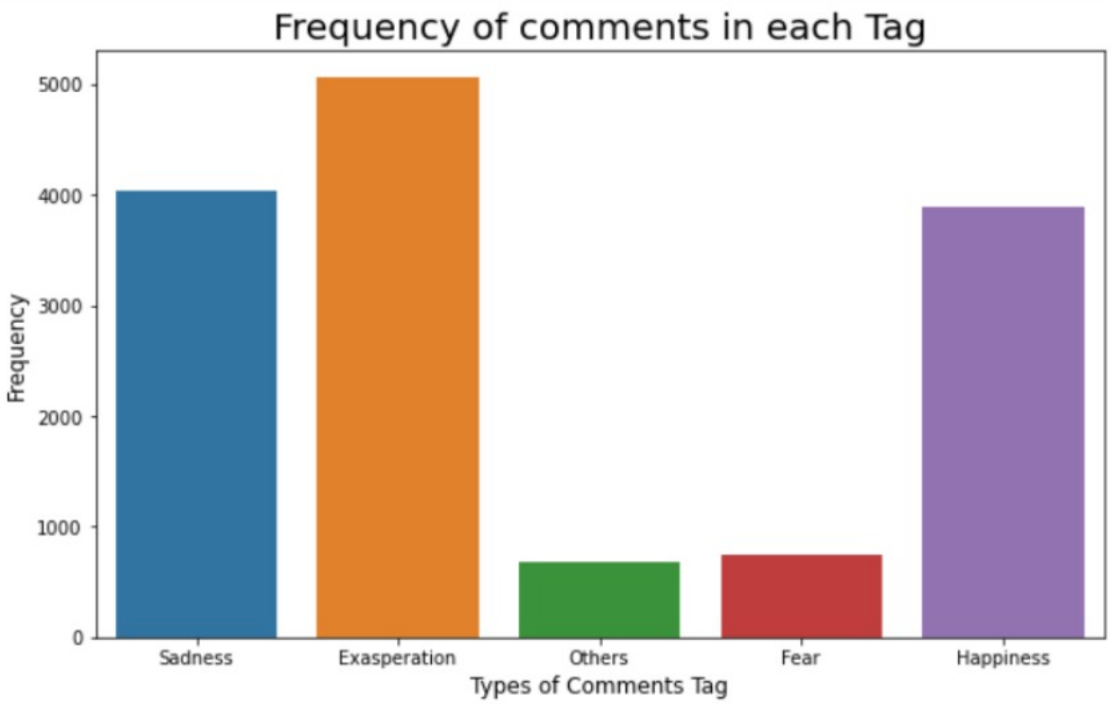

Figure 5.1: Frequency of each class after combination

Description of each class:

Emotion	Total Data
Happiness	3386
Sadness	4035
Exasperation	5057
Fear	752
Others	679

Table 5.2: Number of data in each class after combination

5.1 Multinomial Naive Bayes on Bag of words & TF-IDF

Bag of words is performed to represent the text data into machine understandable language. This is a way to extract features from the text. These features are then fed to the naive Bayes algorithm. After performing Naive Bayes on BoW model we get an accuracy of 61.6%. And the precision, F1, and recall scores are 63.7%, 59.4%, and 61.6% respectively.

	Percision	Recall	F1-score	Support
Exasperation	0.60	0.75	0.66	519
Fear	0.62	0.07	0.12	76
Happiness	0.76	0.69	0.72	395
Others	1.00	0.02	0.04	52
Sadness	0.52	0.55	0.54	399
Accuracy			0.62	1441
Macro-avg	0.70	0.42	0.42	1441
Weighted avg	0.64	0.62	0.59	1441

Table 5.3: Multinomial Naive Bayes Classification Report on BoW representation

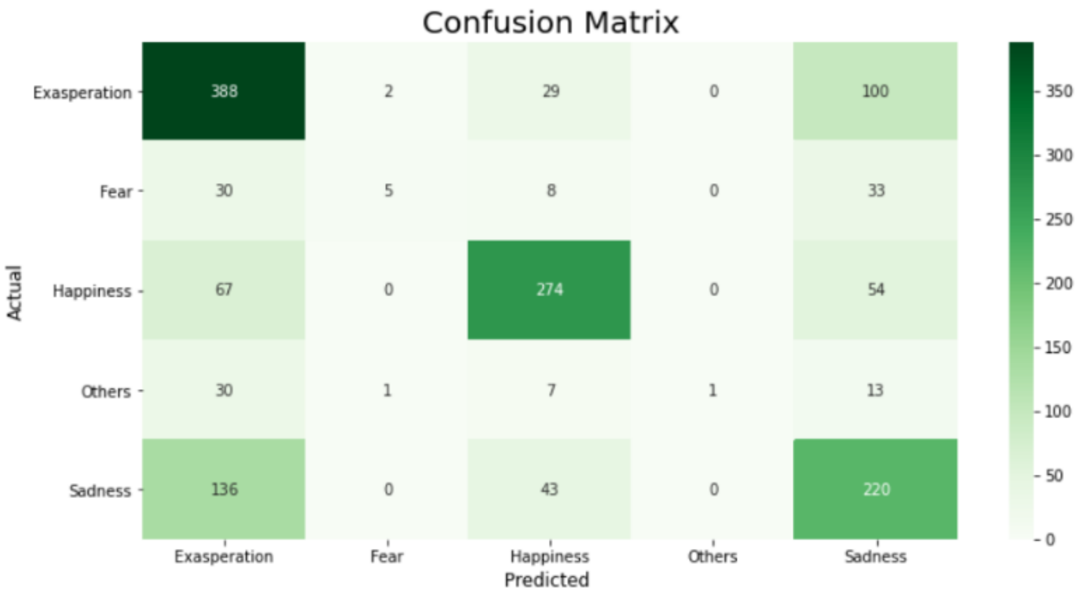

Figure 5.2: Multinomial Naive Bayes Confusion Matrix on BoW Word Representations

TF-IDF is another way to extract features from text data. The TF–IDF approach calculates how essential a word is to the document dataset. TF-IDF uses weights for text mining and information retrieval. The dataset's word frequency determines weight. After performing naive bayes on the TF-IDF model we get an accuracy of 60.9%. And the precision, F1, and recall scores are 57.2%, 57.7%, and 60.9% respectively.

	Percision	Recall	F1-score	Support
Exasperation	0.55	0.83	0.66	519
Fear	0.00	0.00	0.00	76
Happiness	0.80	0.66	0.72	395
Others	0.00	0.00	0.00	52
Sadness	0.56	0.46	0.51	399
Accuracy			0.61	1441
Macro avg	0.38	0.39	0.38	1441
Weighted avg	0.57	0.61	0.58	1441

Table 5.4: Multinomial Naive Bayes Classification Report on TF-IDF representation

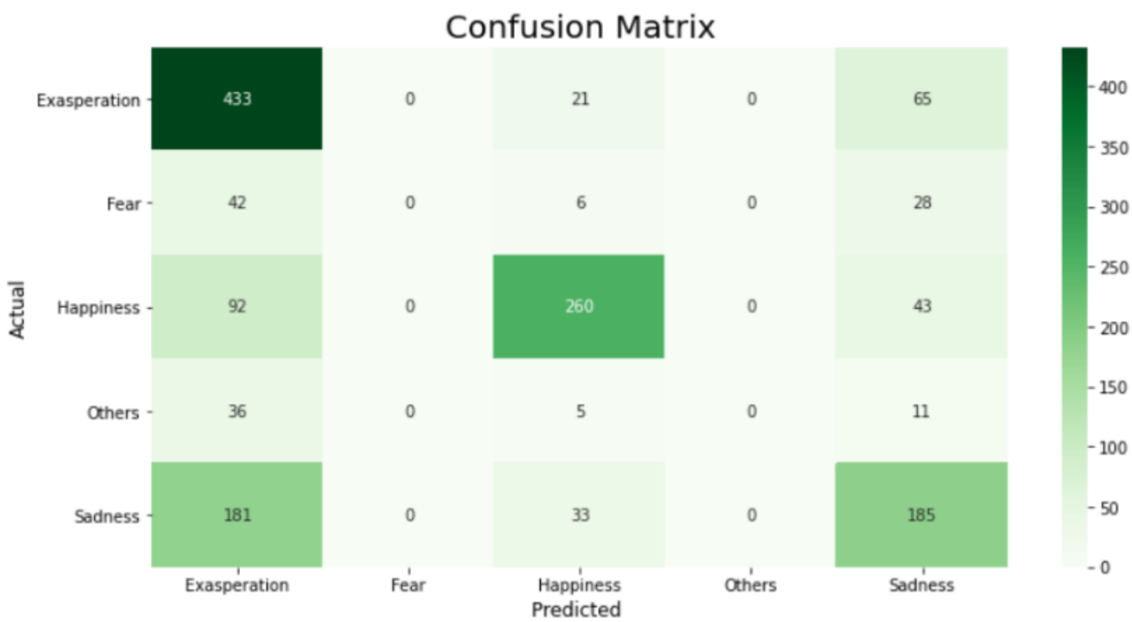

Figure 5.3: Multinomial Naive Bayes Confusion Matrix on TF-IDF Word Representations

Performance comparison Table:

Used Model	Accuracy	Precision	F1 Score	Recall
Naive Bayes (BoW)	61.6%	63.7%	59.4%	61.6%
Naive Bayes (TF-IDF)	60.9%	57.2%	57.7%	60.9%

Table 5.5: Comparison scores of MNB models on different text representation

Here, the Naive Bayes classifier on Bag of Words and TF-IDF 'exasperation', 'happiness' and 'sadness' sentiment gives quite a decent result. But for the 'fear' and 'others' sentiment class the results are poor. Moreover, after applying the Naive Bayes classifier on test data there is a noticeable misclassification is noticed in 'sadness' and 'exasperation' classes.

5.2 Support Vector Machine on Bag of words & TF-IDF

In this classifier, we have used the SVM RBF(Radial Basis Function) kernel with the "gamma" value 0.1 and "c" value of 2000 which gives better accuracy than other kernels and gamma values. SVM provided us with the accuracy of 60.0%. Besides, the precision, F1, and recall scores are 60.0%, 58.6%, and 60.0% respectively.

	Percision	Recall	F1-score	Support
Exasperation	0.59	0.71	0.64	519
Fear	0.61	0.14	0.23	76
Happiness	0.74	0.69	0.71	395
Others	0.43	0.12	0.18	52
Sadness	0.50	0.52	0.51	399
Accuracy			0.60	1441
Macro avg	0.57	0.44	0.46	1441
Weighted avg	0.60	0.60	0.59	1441

Table 5.6: SVM Classification Report on Bag of words Data

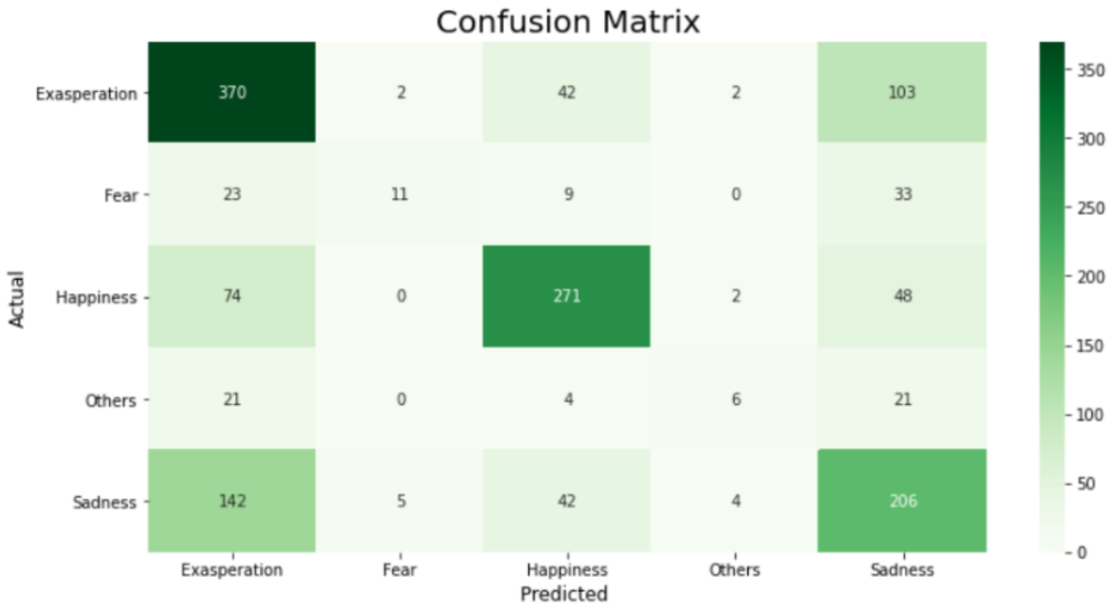

Figure 5.4: SVM Confusion Matrix on BoW Word Representations

On the other hand, in the TF-IDF word representation we get an accuracy of 61.1%. and the precision, F1, and recall scores are 61.3%, 60.1%, and 61.1% respectively.

	Precision	Recall	F1-Score	Support
Exasperation	0.60	0.70	0.65	519
Fear	0.71	0.22	0.34	76
Happiness	0.75	0.70	0.73	395
Others	0.36	0.10	0.15	52
Sadness	0.51	0.54	0.53	399
Accuracy			0.61	1441
Macro avg	0.58	0.45	0.48	1441
Weighted avg	0.61	0.61	0.60	1441

Table 5.7: SVM Classification Report on TF-IDF Data

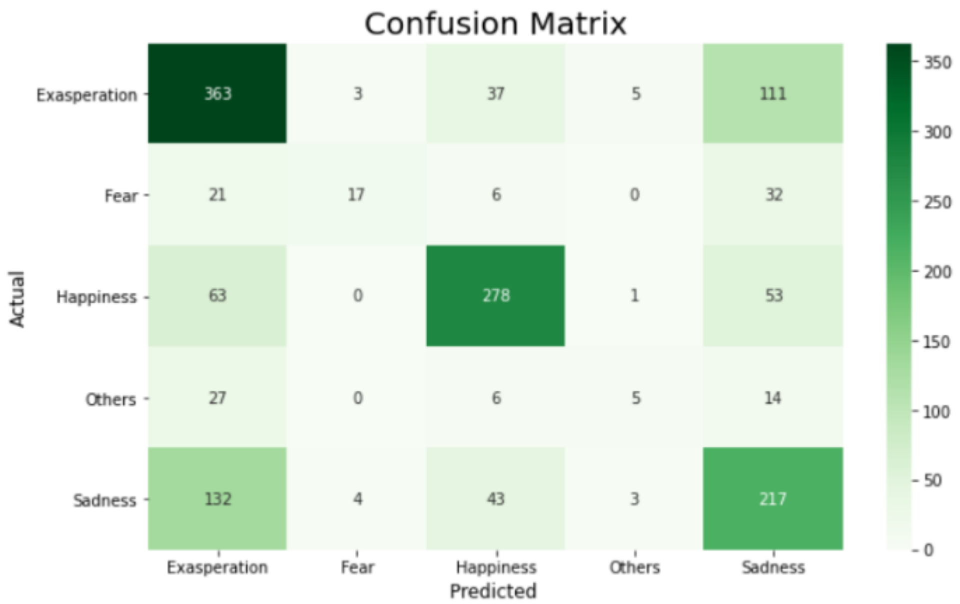

Figure 5.5: SVM Confusion Matrix on TF-IDF Word Representations

Performance comparison Table:

Used Model	Accuracy	Precision	F1 Score	Recall
SVM (BoW)	60.0%	60.0%	58.6%	60.0%
SVM (TF-IDF)	61.1%	61.3%	60.1%	61.1%

Table 5.8: Comparison scores of SVM models on different text representation

Here, the SVM classifier on TF-IDF model performs slightly better for 'fear' sentiment class than the Bag of words model. But All remaining classes' F1 scores are quite similar for both words representing techniques. Furthermore, there is a high ratio of misclassification in the 'sadness' and 'exasperation' classes.

5.3 Recurrent Neural Network

After necessary prepossessing, each sentence was tokenized and padded to a fixed size length of 128. Each word was represented with glove word embedding. The vector length of each word in the glove model is 300. Our proposed neural network model was composed of one embedding layer, one special dropout 1D layer, 2 bidirectional LSTM layers, and finally a dense layer with softmax as an activation function to predict the output. All word vectors are fed into the embedding layer and the weights are initialized with glove embedding weights. The output dimension of the embedding layer is similar to the vector size of the glove model. As optimizers, we selected 'adam' and 'categorical cross entropy' was used to calculate the loss function. In order to reduce overfitting early stopping was used. This is a callback API that provides support to monitor model performance while training. We monitored validation accuracy on the validation set to measure performance during model training. Training will be terminated if there is no validation accuracy improvement after five epochs.

Our LSTM Model provides an accuracy of 60.2%. And the precision, F1, and recall scores are 60.0%, 60.0%, and 60.2% respectively.

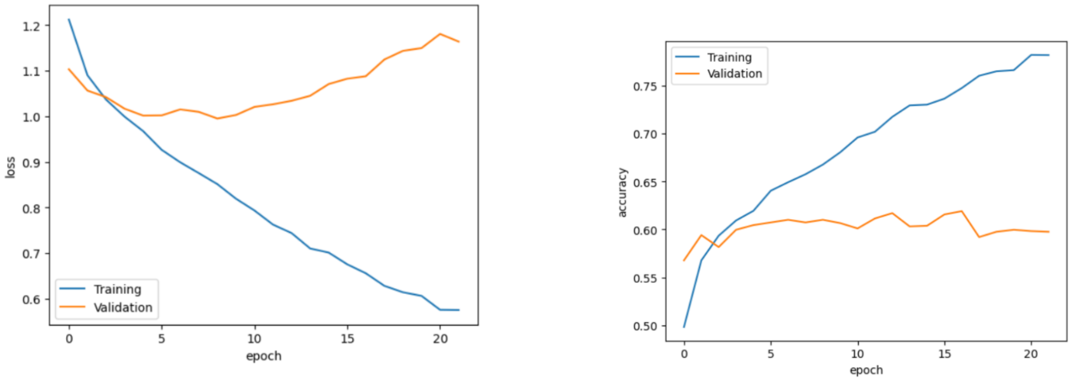

Figure 5.6: LSTM model Loss(Left) and Accuracy(Right)

From the above figure of loss function graph, it can be seen that in the beginning the training and validation loss both are going down but after the 9th epoch the validation loss is continuously increasing. This is the sign of overfitting.

LSTM Performance Table:

Used Model	Accuracy	Precision	F1 Score	Recall
LSTM Model	60.2%	60.0%	60.0%	60.2%

Table 5.9: LSTM performance details

	Precision	Recall	F1-Score	Support
Exasperation	0.62	0.58	0.60	519
Fear	0.49	0.37	0.42	76
Happiness	0.71	0.73	0.72	395
Others	0.27	0.15	0.20	52
Sadness	0.52	0.61	0.56	399
Accuracy			0.60	1441
Macro avg	0.52	0.49	0.50	1441
Weighted avg	0.60	0.60	0.60	1441

Table 5.10: Bidirectional LSTM Classification Report

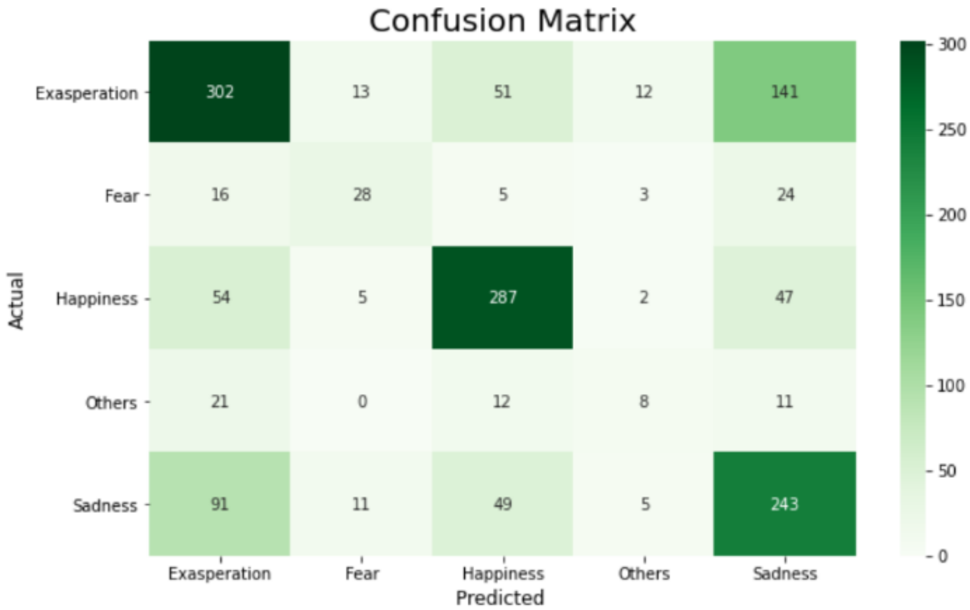

Figure 5.7: LSTM Model Confusion Matrix

In the deep learning approach, in order to handle imbalance class in our data set, weighted class was introduced to reduce weight for the majority class while simultaneously increasing class weight in order to punish the minority class for its incorrect classification. Though, the overall result did not improve. Also, like the previous approaches there is a misclassification between 'exasperation' and 'sadness' class.

5.4 BERT

While implementing BERT, we have used the pre-trained BERT model from 'Bangla-BERT' [25]. We fine-tuned this model for our sentiment classification task. A single vector representation of the whole input sentence must be given to a classifier in order to complete the classification task. For this, all input text was tokenized. Since we are using a pre-trained model for the fine-tune purpose, each token is converted to its corresponding unique IDs. Each sentence was padded to a fixed size length of 128. We fine-tuned the whole model (110 million parameters). Training will be terminated if there is no validation accuracy improvement after five epochs.

Hyperparameter	**Value**
Learning Rate	5e-5
Train_Batch_Size	16
Eval_Batch_Size	8

Table 5.11: Hyperparameters of fine tuned model

Epoch	Training Loss	Validation Loss	Accuracy	Precision	Recall	F1
1	0.717	0.832	0.697	0.656	0.697	0.671
2	0.716	0.845	0.704	0.692	0.704	0.681
3	0.663	0.883	0.696	0.709	0.696	0.698
4	0.242	1.182	0.683	0.711	0.683	0.690
5	0.228	1.400	0.708	0.711	0.708	0.709
6	0.321	1.657	0.700	0.694	0.700	0.696
7	0.084	1.789	0.702	0.702	0.702	0.700
8	0.168	2.027	0.691	0.720	0.691	0.701
9	0.049	1.959	0.703	0.704	0.703	0.707
10	0.001	2.169	0.703	0.699	0.703	0.699

Table 5.12: Training and validation report in each epoch

Eval Loss	1.400
Eval Accuracy	0.707
Eval Precision	0.711
Eval Recall	0.707
Eval F1	0.708

Table 5.13: Validation report

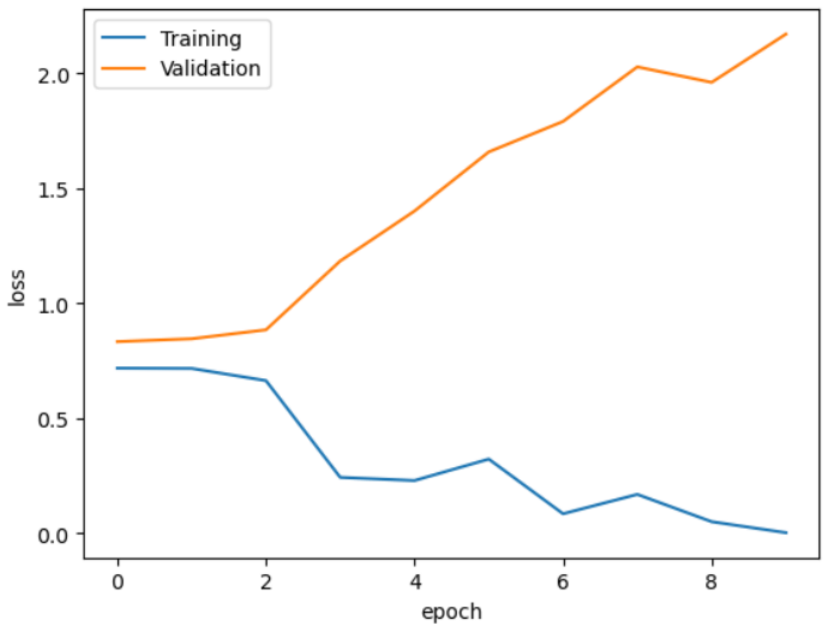

Figure 5.8: BanglaBERT fine-tuned Model Loss

Used Model	Accuracy	Precision	F1 Score	Recall
BanglaBERT	69.2%	70.0%	70.0%	69.2%

Table 5.14: Fine tuned BanglaBERT performance

	Precision	Recall	F1-Score	Support
Exasperation	0.74	0.70	0.72	519
Fear	0.55	0.39	0.46	76
Happiness	0.84	0.83	0.83	395
Others	0.28	0.38	0.32	52
Sadness	0.59	0.65	0.62	399
Accuracy			0.69	1441
Macro avg	0.60	0.59	0.59	1441
Weighted avg	0.70	0.69	0.69	1441

Table 5.15: BanglaBERT fine-tuned model loss

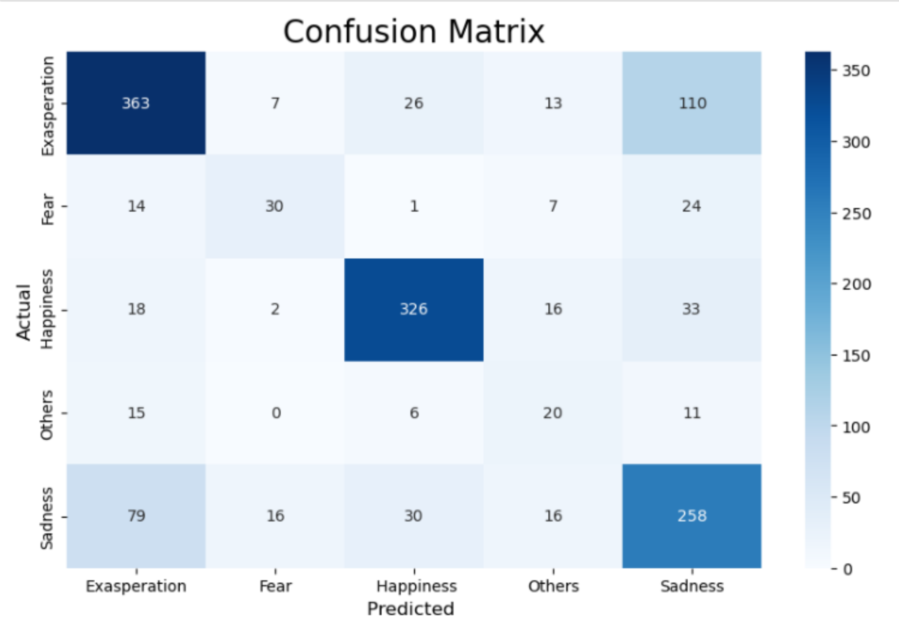

Figure 5.9: BanglaBERT fine tuned model Confusion matrix

Here, The overall performance to classify sentiment is better in transformer models than other models like naive bayes, support vector machine and LSTM. BanglaBERT fine-tuned model performs decently to classify 'exasperation, 'happiness' and 'sadness' class but there is a noticeable misclassification between ''exasperation' and 'sadness' class.

5.5 Result Summary

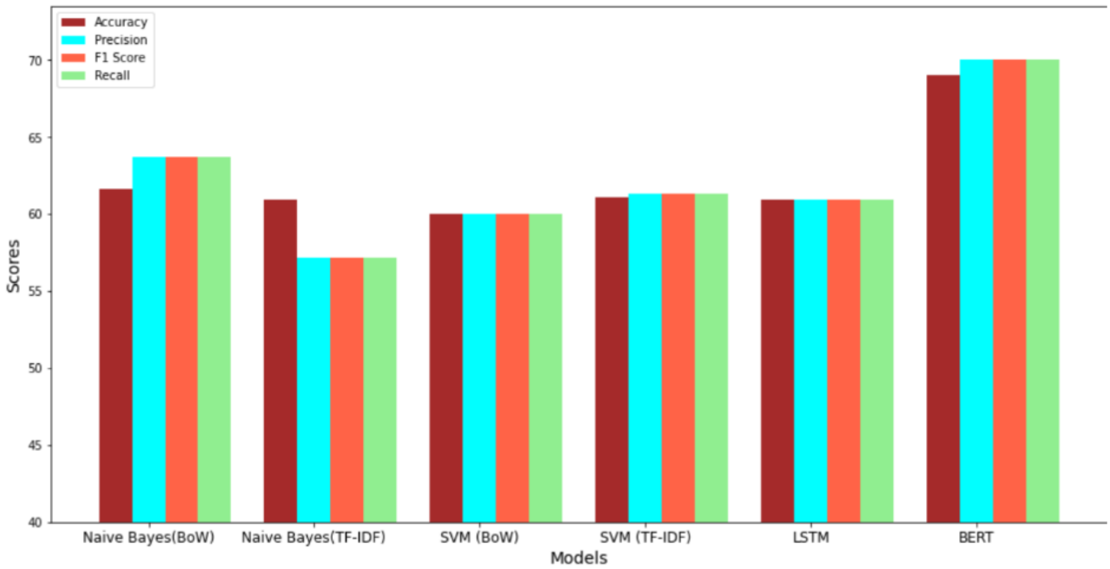

Figure 5.10: Comparison Scores of Implemented Models

From the figure 5.10, By comparing how well these models work, we may conclude that BERT is superior to Naive Bayes, SVM, and LSTM. However, point to be noted that all the models are struggling to detect "fear" and "others" classes. One reason could be that our dataset BANEmo doesn't have sufficient data for these classes. Besides, the "others" class contains "sarcasm" related comments as it was merged with the class later on. Moreover, detecting sarcasm in written text is a critical challenge in NLP as it requires recognizing the contradiction between a statement and its context. Next, "exasperation" and "sadness" statements have very similar semantic meanings thus even human brains sometimes struggle to differentiate between these emotions. Our models were also sometimes struggling to classify between these two classes.

Chapter 6

Conclusion

Sentiment analysis contributes to broader growth by highlighting identifiable qualities shared among speakers of the language. We have created our very own dataset by collecting Bengali text data from social media like Facebook, and Youtube. We also collected text data from the comment sections of different news portals. We believe our dataset represents a real-world scenario where human emotions are expressed disproportionately. By annotating manually, we are able to discern the category of the emotion that the Bengali texts express. The models are taught to detect patterns and differentiate between different basic emotions like happiness, sadness, disgust, fear, surprise, and anger by using supervised ML and pre-labeled data. While working on this, we experimented with a number of methods for extracting the most useful information from the dataset. Here, we used Bag of Words and TF-IDF for pre-processing of data. Additionally, we have used some different kinds of word embedding techniques such as Glove, and BERT embedding to represent text data understandable for machines. For measuring the performance, we have used supervised machine learning models like Naive Bayes, and Support Vector Machine. Furthermore, we used deep learning techniques like LSTM, and Transformers (BERT).
By analyzing the performances of these models we can deduce that the BERT model outperforms Naive Bayes, SVM, and LSTM models comfortably.

6.1 Limitations

We get outcomes that were comparatively better than others if we take into account the total number of classes. In addition, our dataset was based on real-world instances in which text data was disproportionately distributed among classes. In addition, there have not been many prior works on categorizing Bangla text using both machine learning and deep learning techniques. However, there are a variety of different feature selection techniques used in other languages that we did not employ. Our dataset only contains textual information from Facebook, YouTube, and the comment areas of various online news sites. There are numerous additional sorts of data in Bengali, including periodicals, literature, novels, and religious texts, among others. The set of data could have been more diverse. Besides, We think there might be some inconsistency with the annotations, which is why the model might disagree more in some cases. Moreover, our model was struggling to detect sarcasm. Sarcasm identification is a crucial natural language processing challenge.

Understanding this requires recognizing the contradiction between a statement and its context. Next, as a morphologically rich language, Bangla presented challenges when we attempted to implement certain features. Due to the lack of high-quality Bangla stemming and lemmatization methods, we have been unable to operate as efficiently as we would want.

6.2 Future work

We hope to improve sentiment and emotion identification in the future by incorporating more features and topical data. Our machine learning and deep learning approach will yield more accurate predictions if we employ a larger, more evenly distributed, and more diverse dataset than we currently have. To do so, we need to compile an enormous quantity of sentimental data. In the future, we hope to test our approach on even more data sets. As there are very few emotion detection Bengali datasets available publicly, we plan to make our dataset available publicly to help future researchers. Before that, we plan to cross-validate our dataset with the help of Cognitive and linguistic experts in the Bengali language.

Furthermore, we will try to improve its performance by developing a hybrid mechanism using these supervised machine learning models and deep learning techniques like Naive Bayes, Support Vector Machine, LSTM, and Transformers (BERT).We can also implement the latest Transformer models to achieve better results. Besides, we will try to make a useful application for users by implementing these concepts.

Bibliography

[1] T. Dalgleish **and** M. Power, *Handbook of cognition and emotion.* John Wiley & Sons, 2000.

[2] E. Cambria **and** B. White, Jumping NLP curves: A review of natural language processing research, *IEEE Computational intelligence magazine*, **jourvol** 9, **number** 2, **pages** 48–57, 2014.

[3] R. E. Jack, O. G. Garrod **and** P. G. Schyns, Dynamic facial expressions of emotion transmit an evolving hierarchy of signals over time, *Current biology*, **jourvol** 24, **number** 2, **pages** 187–192, 2014.

[4] F. Sadeque, T. Solorio, T. Pedersen, P. Shrestha **and** S. Bethard, Predicting continued participation in online health forums, **in***Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis* 2015, **pages** 12–20.

[5] C. Zhou, C. Sun, Z. Liu **and** F. Lau, A C-LSTM neural network for text classification, *arXiv preprint arXiv:1511.08630*, 2015.

[6] J. L. Ba, J. R. Kiros **and** G. E. Hinton, Layer normalization, *arXiv preprint arXiv:1607.06450*, 2016.

[7] A. Hassan, M. R. Amin, A. K. Al Azad **and** N. Mohammed, Sentiment analysis on bangla and romanized bangla text using deep recurrent models, **in***2016 International Workshop on Computational Intelligence (IWCI)* IEEE, 2016, **pages** 51–56.

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

[9] A. P. Jain **and** P. Dandannavar, Application of machine learning techniques to sentiment analysis, **in***2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)* IEEE, 2016, **pages** 628–632.

[10] A. Vaswani, N. Shazeer, N. Parmar **andothers** Attention is all you need, *Advances in neural information processing systems*, **jourvol** 30, 2017.

[11] J. Devlin, M.-W. Chang, K. Lee **and** K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805*, 2018.

[12] M. Rathi, A. Malik, D. Varshney, R. Sharma **and** S. Mendiratta, Sentiment analysis of tweets using machine learning approach, **in***2018 Eleventh international conference on contemporary computing (IC3)* IEEE, 2018, **pages** 1–3.

[13] N. I. Tripto **and** M. E. Ali, Detecting multilabel sentiment and emotions from bangla youtube comments, **in** *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)* IEEE, 2018, **pages** 1–6.

[14] R. Tudu, S. Saha, P. N. Pritam **and** R. Palit, Performance analysis of supervised machine learning approaches for bengali text categorization, **in** *2018 5th Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)* IEEE, 2018, **pages** 221–226.

[15] M. A.-U.-Z. Ashik, S. Shovon **and** S. Haque, Data set for sentiment analysis on Bengali news comments and its baseline evaluation, **in** *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)* IEEE, 2019, **pages** 1–5.

[16] M. Rahman, M. Seddiqui **andothers**, Comparison of classical machine learning approaches on Bangla textual emotion analysis, *arXiv preprint arXiv:1907.07826*, 2019.

[17] F. Sadeque, S. Rains, Y. Shmargad, K. Kenski, K. Coe **and** S. Bethard, Incivility detection in online comments, **in** *Proceedings of the eighth joint conference on lexical and computational semantics (* SEM 2019)* 2019, **pages** 283–291.

[18] F. Y. Sadeque, User Behavior in Social Media: Engagement, Incivility and Depression, phdthesis, The University of Arizona, 2019.

[19] R. C. Staudemeyer **and** E. R. Morris, Understanding LSTM–a tutorial into long short-term memory recurrent neural networks, *arXiv preprint arXiv:1909.09586*, 2019.

[20] G. Xu, Z. Yu, H. Yao, F. Li, Y. Meng **and** X. Wu, Chinese text sentiment analysis based on extended sentiment dictionary, *IEEE Access*, **jourvol** 7, **pages** 43 749–43 762, 2019.

[21] M. A. Alonso, D. Vilares, C. Gómez-Rodríguez **and** J. Vilares, Sentiment analysis for fake news detection, *Electronics*, **jourvol** 10, **number** 11, **page** 1348, 2021.

[22] D. Bhimani, R. Bheda, F. Dharamshi, D. Nikumbh **and** P. Abhyankar, Identification of Hate Speech using Natural Language Processing and Machine Learning, **in** *2021 2nd Global Conference for Advancement in Technology (GCAT)* IEEE, 2021, **pages** 1–4.

[23] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal **and** M. Islam, Bangla text sentiment analysis using supervised machine learning with extended lexicon dictionary, *Natural Language Processing Research*, **jourvol** 1, **number** 3-4, **pages** 34–45, 2021.

[24] P. Chandra **and** U. Prasad, Classification of Emojis using Artificial Neural Network and Natural Language Processing, **in** *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)* IEEE, 2021, **pages** 205–212.

[25] A. Bhattacharjee, T. Hasan, W. A. Uddin **andothers**, Banglabert: Lagnuage model pretraining and benchmarks for low-resource language understanding evaluation in bangla, *Findings of the North American Chapter of the Association for Computational Linguistics: NAACL*, 2022.

[26] M. Sakib Ullah Sourav **and** H. Wang, Transformer-based Text Classification on Unified Bangla Multi-class Emotion Corpus, *arXiv e-prints*, arXiv–2210, 2022.