Emotion Detection from Bengali Textual Data Using Machine Learning

By

Aninda Das Showmik
ID – 182001612
&

Samia Tazin
ID – 182002512

Supervised by

Dr. Mohammed Nazim Uddin
Professor & Associate Dean
SoSET, East Delta University

Department of Computer Science & Engineering

School of Science, Engineering & Technology

East Delta University, Noman Society, East Nasirabad, Chittagong-4209

The undergraduate thesis titled **"Emotion Detection From Bengali Textual Data using Machine Learning"** submitted by **Aninda Das Showmik (ID-182001612) & Samia Tazin (ID-182002512)** has been accepted as satisfactory in fulfillment of the requirement for the degree of Bachelor of Science (B.Sc.) in Computer Science and Engineering (CSE) to be awarded by East Delta University.

Board of Examiners

Dr. Md. Ishtiaque Aziz Zahed **Chairman**
Associate Professor
School of Science, Engineering and Technology
East Delta University

Linkon Chowdhury **Member**
Assistant Professor
School of Science, Engineering and Technology,
East Delta University

Sohrab Hossain **Member**
Assistant Professor
School of Science, Engineering and Technology
East Delta University

Dr. Mohammed Nazim Uddin **Supervisor**
Professor and Associate Dean
School of Science, Engineering and Technology
East Delta University

Declaration

It is hereby declared that the contents of this project are original and any part of it has not been submitted elsewhere for the award of any degree or diploma.

Date :

Aninda Das Showmik
182001612

Samia Tazin
182002512

Acknowledgments

First, I want to thank our respective supervisor Prof. Dr. Mohammed Nazim Uddin give us instructions and valuable time. Without his guidance, we could never have done this. His help makes my way much more straightforward.

Then, I want to thank Dr. Ishtiaque Aziz Zahed, Linkon Chowdhury, and Sohrab Hossain sir for providing feedback after reviewing the thesis.

At last, I would very much like to thank my family who has always got my back and some seniors who helped me during the thesis development in many sectors.

Table of Contents

List of Figures

List of Tables

List Of Abbreviations

AI	**:**	Artificial Intelligence
NLP	**:**	Natural Language Processing
ML	**:**	Machine Learning
DL	**:**	Deep Learning
BEmoC	**:**	Bengali Emotional Corpus
BYCD	**:**	Bengali YouTube Comments Dataset
EBEmoD	**:**	Extended Bengali Emotion Dataset)
CNN	**:**	Convolutional Neural Network
BiLSTM	**:**	Bidirectional Long Short-Term Memory
SVC	**:**	Support Vector Classifier
MNB	**:**	Multinomial Naïve Bayes
LR	**:**	Logistic Regression
DT	**:**	Decision Tree
RF	**:**	Random Forest
VC	**:**	Voting Classifier

Abstract

Emotion detection refers to the identification of emotion from contextual data. Detecting emotion is one of the most challenging cases in the automated understanding of language. Although various types of research on emotion classification progressed in high resource languages like English, French, Arabic and Chinese. Research on the Bengali language is still in its infancy stages. Few types of research have been done on detecting emotions in the Bengali language. The deficiency of benchmark corpora and the unavailability of natural language processing tools make it more complicated. This work explores and investigates the comparison between the outcome of six machine learning (SVC, LR, DT, MNB, RF and Voting) including two ensemble-based techniques to classify textual emotion into three classes: sad, joy and anger. A dataset containing 4500 Bengali texts is formed to carry out emotion categorization. This work explores and investigates the comparison between the outcomes of the standard classifier models. Experimental outcomes show that the Multinomial Naïve Bayes surpasses other techniques which achieved the highest accuracy on the test data for word Uni-gram but using Bi-gram and Tri-gram the Voting classifier achieved the highest accuracy.

Keywords: Emotion detection, Emotion classification, Machine learning, Emotion categorization.

Chapter 1

Introduction

1.1 Background of the study

Now is the age of 'Information and Technology' that is making everything going forward to our day to day like easier. Emotion detection from textual dialogues is text-based sentiment analyzing process, one of the fastest growing branches of NLP (Natural Language Processing). To make human- machine interaction smoother, NLP has make it through and become a popular field. Thanks to the advancement of technology, an extensive amount of people now use social media as a platform to interact with each other and express their opinions, emotions and experiences through outlooks tweets, posts, comments, basically in a form of text. It represents various peoples' outlooks on various issues such as business, politics, sports, economy and so on [1]. This vast data and information have encouraged emotion and sentiment analysis research for enterprises, consumers, organization operators and so on. Numerous studies are being carried out in sentiment analysis for emotion detection as there are plenty of sources of data and its benefits of deliverable offers. Developing automated systems or NLP tools for Bengali language is a rising demand that can comprehend one's emotion and reaction based on an event, product, and situation and so on.

According to Ekman [2], six basic human emotions are there based on facial traits: happiness, fear, anger, sadness, disgust and surprise. The text expression can also represent these emotions. Emotion detection has been observed mostly in English, Arabic, French & other high-resource languages but very few studies have investigated emotion in under-resourced languages like Bengali because of the scarcity of resources & deficiency of standard corpora. Bengali is the seventh most spoken language in the world with closely 228 million native speakers. More than six crores social media users' digital technology and easy access to the internet brought about extensive data in the Bengali language. Many Bengali- speaking people, who have limited skill in English face struggles to employ other resources of the language of English. Many researches and efforts are ongoing to make it easy to use Bengali in various technical domains. Extracting and identifying emotions from this textual data is a challenge due to the lack of Bengali language processing tools, standard corpora and limited sources. Most of the work on Bengali texts focused on binary sentiment analysis where positive is represented as happiness emotion and negative is represented as sadness emotion. But that is not adequate for identifying the exact emotion from textual data.

1.2 Objective of the study

This work aims to contribute to creating a Bengali emotion corpus containing 3951 texts with suitable annotation to classify each text into the three emotion classes: sad, joy, anger. The main label of text had done by an expert to make the dataset more accumulate. There will apply word n -gram, TF-IDF techniques for feature extraction. This work tells of an investigation of the outcome of ML approaches on the corpus and a comparison of the performances with the experimental validation of the corpus.

1.3 Outline of the study

This work can be split up into five sections. Related study of this work is illustrated in Section 2, where various approaches have been taken such as AI, ML etc. Proposed methodology of this work is described in section 3. Section 4 gives the experimental results and analysis achieved from the work. In the end, Section 5 draws the conclusion of the study and future work.

Chapter 2

Literature Review

In this chapter, we will discuss different research fields of Artificial Intelligence for Bengali text sentiment analysis.

By the end of this chapter, there will be also mentions of whose works influence us.

2.1 AI for Sentiment Analysis

Sentiment analysis by artificial intelligence is one of the most well-liked topics in the computer science research area. Because it is particularly effective for obtaining customer feedback or public reactions.
Various social media have become powerful means to express people's opinions. So, emotion detection from texts using AI has become a an interesting and vast area of NLP(Natural Language Processing) research. It can be used in various areas such as data mining, recommendation systems, human- computer interaction, understanding expressed emotion, psychology and so on[3].
AI can detect the sentiment of any data but first, the technology needs to be properly taught. We used roughly 4000 data for training.

2.1.1 Emotion Detection from Bengali Textual Data

One of the major uses of artificial intelligence (AI) and Natural Language Processing (NLP) is emotion recognition from texts. It is a very important field for research because it is the way, how a machine can understand human emotions. There are many works has been done with many types of language Like English, Spanish, French, Arabic etc. Bengali is the seventh most spoken language by the total number of speakers in the world [4] . Some works also have been done with our Bengali language.

2.1.2 ML for Emotion Detection from Bengali Textual Data

A significant wing of artificial intelligence which is commonly used for sentiment analysis is machine learning. There are many supervised machine learning models which are very popular for text analysis. Like support vector machine, Logistic regression, K- nearest algorithm etc. There are some works have been done to detect emotion from Bengali texts. Many of them were applied to detect binary emotions (Positive, Negative). There are very little numbers of works with more than 2 emotions. Sara and Kingshuk worked with 3 classes [5] (Happy, Sad, Angry) and gets 78.6% accuracy by applying the Naïve Bayes model. They worked with only a thousand data and applied only one model. Avishek et al. [6] trained the system with approx. 9000 data for 6 classes and applied ML, DL and Transformer Model. By applying 4 ML models, they got the highest 60.75% f1 score for logistic regression. Tanzia et al. [7] trained the system with approx. 9000 data for 6 classes and applied ML, DL and Transformer Model. By applying 22 standard classifier models developed based on three deep learning techniques, they got the highest weighted f1-score of 62.46% (for EBEmoD) and 67.57% (for BYCD), respectively using the ensemble of CNN and BiLSTM.

2.1.3 Ensemble for Sentiment analysis From Bengali Textual Data

An ensemble model is an approach for classifying data which is combine two or more classifier models to get a more accurate and obvious result. Assessment of emotion detection with ensemble models from Bangla text is so infrequent. So we take some works as reference, those are different in language but done with ensemble techniques. Nádia et al. [8] Introduce an approach that automatically classifies the sentiment of tweets by using classifier ensembles and lexicons. They applied a voting and averaging ensemble approach. To improve the result they applied the ensemble technique on (Multinomial Naive Bayes, SVM, Random Forest, and Logistic Regression) 4 basic ML classifier models. They had improved the average f1 -score from 65.95 to 78.25. Gang Wang et al. [9] compare and evaluate the effectiveness of three well-known ensemble methods Bagging, Boosting, and Random Subspace based on five base learners Naive Bayes, Maximum Entropy, Decision Tree, K Nearest Neighbor, and Support Vector Machine for sentiment classification.

2.2 Developing Corpus of Bengali Texts

Asif et al. [10] developed a corpus (hereafter called BEmoC) of 7000 texts. They did the development process consisting of 4 steps: data crawling, pre-processing, labeling, and verification. They labeled all text with basic emotion categories such as anger, fear, surprise, sadness, joy, and disgust, respectively. They evaluated the dataset by calculating the Cohen K score of 0.969. Moreover, they assessed the produced BEmoC using more metrics, such as Zipf's law, coding accuracy, the term that describes emotions most frequently, and the density of emotions.

Chapter 3

Methodology

The needed components and procedure of the experiment is described in this chapter. In this section, our proposed approach is described that will be applied on our preprocessed dataset to acquire experimental results.

There are 7 steps need to complete our whole process. They are: data collection, pre-processing and data validation, data labeling, Feature extraction, ML classifiers, Result analysis, performance evaluation.

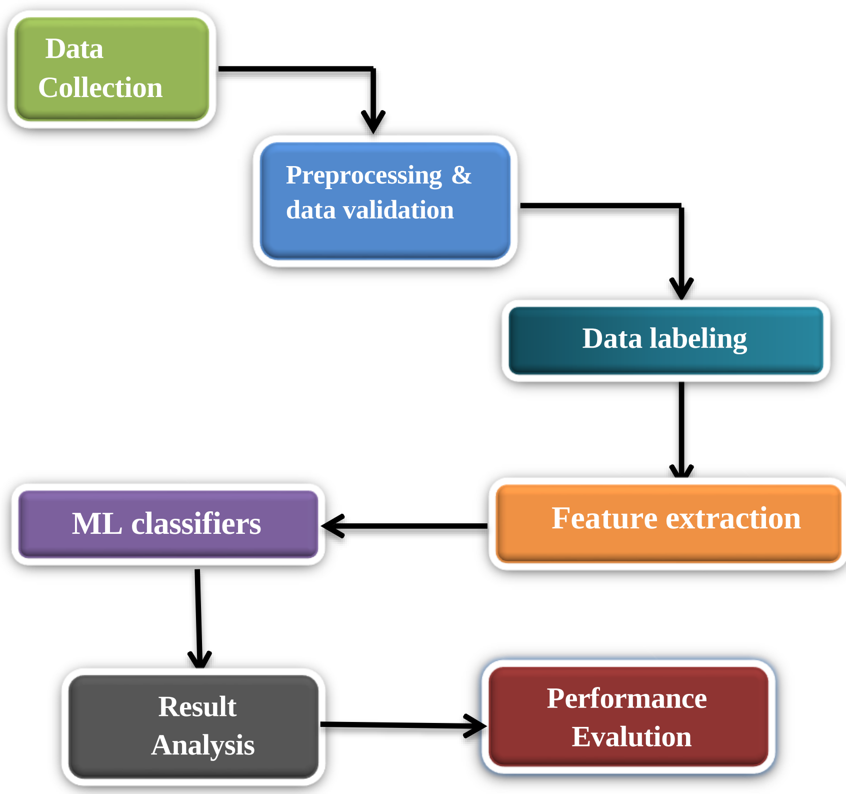

Figure 3-1: Architecture of the proposed approach

The architecture of the proposed approach is shown in figure 3.1. The first needed element is the standard corpora on Bengali texts. Due to the scarcity of standard datasets we build a dataset on Bengali textual data on our own. The available datasets were very noisy and had data of mixed emotions.

After developing the dataset, we preprocess the raw collected data and label suitable emotion class on data which will be done by an expert. Then, after applying feature extraction techniques (Word n-grams, TF-IDF), we apply various ML classifier models that will give us an estimated outcome and at last we evaluate the performance between all models.

3.1 Dataset

The dataset developed and used for this work contains a large number of user comments from different social media posts, videos, you tube comments and public posts of popular blogs. You tube comments were harvested in automated way using API [11]. The development procedure of the dataset was adopted from the guidelines at [12]. The data was accumulated selectively and only collected when it supported the definition of emotion according to [2]. Initially 4500 data were collected.
Source wise distribution for collecting data shown in Figure 3.2:

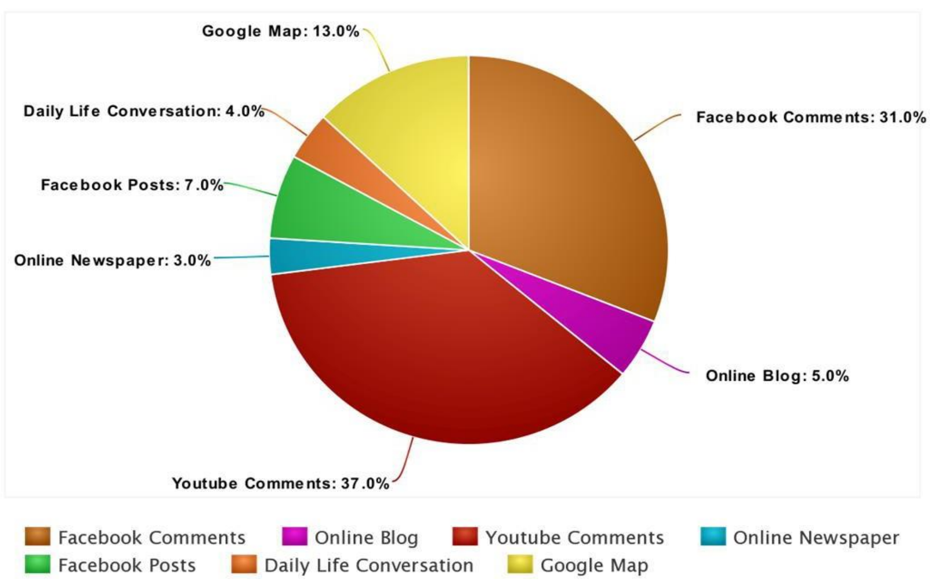

Figure 3-2: Sources for Data Collection

3.1.1 Preprocessing

Data preprocessing is a must before applying any machine learning algorithms to data. It is a required step before going any further. Algorithms learn from these data, so these data must be cleaned and noise-free because the learning outcome of the algorithms depends on proper data. After data collection, the required step is preprocessing the data. Initially, 4500 data were collected.

Raw data can have errors, duplications, noises and unnecessary information. Preprocessing is required which removes noises from data and helps achieve accurate analytical results.

Remove_noise() built-in function from BnPreprocessing() [13] library is used for the initial preprocessing of the Bengali textual documents. The processing techniques are as follows:

1. Removal of punctuations and digits.
2. Removal of emoticons/emojis
3. Removal of stop words.
4. Removal of URLs.
5. Removal of HTML tags.
6. Removal of English words.

Any kind of unnecessary dot, comma, emoticon, hyphens and other symbols are removed. All the emoticons are removed since we aim to consider textual data only. Remove_noise() function remove emoticons also but some emojis remained. There demoji() library function needed to remove those. Hence Stops words (অথচ, অথবা, এবং, হয়) are filtered and removed. After preprocessing, the sample data are given in Figure 3.3.

index	Text	preprocessed_text
4344	ভালো করেছে সে। তাকে সমর্থন করছি।	ভালো সমর্থন করছি
4347	বাস্তবে সে একজন সুপার হিরো	বাস্তবে একজন সুপার হিরো
4348	❤️❤️❤️sabas..❤️	
4352	❤️❤️❤️	
4353	আমাদের দেশে যেখানে টাকার জন্য পিতাকে পিটিয়ে জখম,খুন,মাকে গুলি করে হত্যা করা হচ্ছে সেখানে আরেক দেশে পিতাকে ভালোবেসে হাতে অস্ত্র তুকে নিচ্ছে।স্যালুট ভাই।আল্লাহ আপনার সহায় হোন।আমিন।	দেশে টাকার পিতাকে পিটিয়ে জখম খুন হত্যা আরেক দেশে পিতাকে ভালোবেসে অস্ত্র তুকে নিচ্ছে স্যালুট ভাই আল্লাহ সহায় হোন আমিন
4354	বাংলাদেশেও ব্যাঙ্ক থেকে টাকা তুলতে গেলে এই অবস্থা হবে। কোন ব্যাংকে টাকা নেই।	বাংলাদেশেও ব্যাঙ্ক টাকা তুলতে অবস্থা ব্যাংকে টাকা
4355	গর্বিত সন্তান	গর্বিত সন্তান
4357	অনেক কষ্টে এই সিদ্ধান্ত নিছে যাই হোক এমন সন্তান সবার ভাগ্যে জুটে না	কষ্টে সিদ্ধান্ত নিছে যাই সন্তান ভাগ্যে জুটে
4360	আরবরা মহান জাতি	আরবরা মহান জাতি
4365	ভালো মানুষ এই জন্যই খারাপ হয়ে যায়	ভালো মানুষ জন্যই খারাপ হয়ে যায়
4367	এখন সে পুরা বিশেব র শেরা	পুরা বিশেব শেরা
4368	হাত সে যত দূর্বলই হোক, ক্রোধ তাতে শক্তির সঞ্চার করে💪💪💪	দূর্বলই ক্রোধ শক্তির সঞ্চার
4369	আমাদের দেশে এমন হলে, একাধিক টকশো এখন পর্যন্ত হয়ে যেত।।।	দেশে একাধিক টকশো

Figure 3-3: Sample data after preprocessing

3.1.2 Data Validation

Data validation is much necessary for a new dataset. Data validation was done in the following steps:

- **Removal of empty rows:** After preprocessing the data, some unnecessary information were discarded hence some rows turned empty. Those empty rows were removed.
- **Removal of duplicate texts:** Raw data can have duplicate data and noises. Duplicate data can influence the possible outcome so they were discarded.
- **Removal of texts of unexpected length:** After preprocessing, some data turned into very short texts in length. Length of texts less than three words were discarded to get proper outcome.

Sample data after validation can be seen in Figure 3.4.

index	Text	preprocessed_text
4344	ভালো করেছে সে। তাকে সমর্থন করছি।	ভালো সমর্থন করছি
4347	বাস্তবে সে একজন সুপার হিরো	বাস্তবে একজন সুপার হিরো
4353	আমাদের দেশে যেখানে টাকার জন্য পিতাকে পিটিয়ে জখম,খুন,মাকে গুলি করে হত্যা করা হচ্ছে সেখানে আরেক দেশে পিতাকে ভালোবেসে হাতে অস্ত্র তুকে নিচ্ছে।স্যালুট ভাই।আল্লাহ আপনার সহায় হোন।আমিন।	দেশে টাকার পিতাকে পিটিয়ে জখম খুন হত্যা আরেক দেশে পিতাকে ভালোবেসে অস্ত্র তুকে নিচ্ছে স্যালুট ভাই আল্লাহ সহায় হোন আমিন
4354	বাংলাদেশেও ব্যাঙ্ক থেকে টাকা তুলতে গেলে এই অবস্থা হবে। কোন ব্যাংকে টাকা নেই।	বাংলাদেশেও ব্যাঙ্ক টাকা তুলতে অবস্থা ব্যাংকে টাকা
4357	অনেক কষ্টে এই সিদ্ধান্ত নিছে যাই হোক এমন সন্তান সবার ভাগ্যে জুটে না	কষ্টে সিদ্ধান্ত নিছে যাই সন্তান ভাগ্যে জুটে
4360	আরবরা মহান জাতি	আরবরা মহান জাতি
4365	ভালো মানুষ এই জন্যই খারাপ হয়ে যায়	ভালো মানুষ জন্যই খারাপ হয়ে যায়
4367	এখন সে পুরা বিশেব র শেরা	পুরা বিশেব শেরা
4368	হাত সে যত দূর্বলই হোক, ক্রোধ তাতে শক্তির সঞ্চার করে💪💪💪	দূর্বলই ক্রোধ শক্তির সঞ্চার
4369	আমাদের দেশে এমন হলে, একাধিক টকশো এখন পর্যন্ত হয়ে যেত।।।	দেশে একাধিক টকশো
4370	বিপদের সময় কাজে লাগানোর জন্য মানুষ ব্যাংকে টাকা জমিয়ে রাখে,আর সে যখন বিপদের সময় টাকা না পাবে সেত অন্যায় করতে বাধ্য হবেই,আমার মতে বাবাকে বাঁচানোর জন্য ওনি ঠিক কাজই করেচে,বাবার কিছু হলে ব্যাংক বাবাকে পিরিয়ে দিতে পারবে,তখন ওনি আরো ভয়ানক হতো	বিপদের সময় লাগানোর মানুষ ব্যাংকে টাকা জমিয়ে রাখে বিপদের সময় টাকা পাবে সেত অন্যায় বাধ্য হবেই মতে বাবাকে বাঁচানোর ওনি কাজই করেচে বাবার ব্যাংক বাবাকে পিরিয়ে পারবে ওনি আরো ভয়ানক হতো
4378	খুব ভালো লাগল। উচিৎ কাজ করছে।	ভালো লাগল উচিৎ
4387	ব্যাংকে আমার টাকা তুলতে গেলে ঘন্টার পর ঘন্টা দাড়িয়ে থাকতে হয়	ব্যাংকে টাকা তুলতে ঘন্টার ঘন্টা দাড়িয়ে থাকতে
4391	এ দিকে আমাদের দেশে বাবা কে পিটিয়ে ৩১ লক্ষ টাকা নিয়ে নেয়	দেশে বাবা পিটিয়ে টাকা নেয়

Figure 3-4: Validated Sample Data

3.1.3 Data Labeling

Data labeling is an essential and very important process of identifying data and providing context which makes the machine learning models learn from it. Labeling data correctly allows machine learning models to make decisions accurately.
Initially around 4500 Bengali textual data were collected. After preprocessing, total 3951 Bengali texts were considered for the dataset. As we followed the supervised machine learning technique, we labeled the dataset based on the presence of words in addition to phrases corresponding to emotional content as well as the overall emotion lying in that

Comment. To identify various unique characteristics of emotion classes in Bengali texts, several attributes [10] are considered such as

- ***Emotion keywords:*** There are some terms or words that are often used to express emotion. For example, "খুশি" , "সুন্দর" these words are used to express the emotion class Joy. "রাগ" , "শবরক্ত" these words often represent the emotion class anger. So these kinds of specific words are used to express each emotion class.

- ***Intensity of emotion words:*** Specific emotion words represents different emotion classes based on the context. Even though single specific word can represent a particular emotion, it also can represent another emotion on the basis of intensity of that specific emotion words.

- ***Semantics of sentence:*** it is one of the core characteristics of a sentence. It plays a vast role to identify precise emotion class of a given statement. Assigning right labels of emotions vastly depends on the semantics of the texts. It deals and finds how words are used in a sentence and what context they represent.

- ***Emotion engagement:*** while reading a texts, annotator can feel the emotion reflected from the text. For example, " নতুন বছরর খুব সুন্দর একটা উপহার পেলাম খুব ভাল লাগল", the annotator can feel a great pleasure or joy while reading this sentence, so the expression represents Joy.

We labeled the corpus in two ways. First, the annotators initially labeled the text of corresponding preprocessed and cleaned data separately and they were labeled without having any prejudiced feeling towards any specific religion, customs.

To assign the suitable label, the final labeling process was done by an annotator who has a good proficiency in Bengali language (Graduate in Bengali and doing MA. In Bengali). Figure 3.5 represents the sample data after labeling.

index	Text	preprocessed_text	Us	Expert
2530	সংবাদিকা ও তথাকথিত সামাজিক যোগাযোগ মাধ্যম এর কাছে আমরা সবাই দিন দিন জিম্মি হয়ে যাচ্ছি।	সংবাদিকা তথাকথিত সামাজিক যোগাযোগ সবাই জিম্মি যাচ্ছি	Sad	Sad
2531	এত ছোট বিষয় নিয়ে কত কাহিনী, এর চেয়ে যে কত বড় জঘন্য কাজ হচ্ছে তার কিছু হচ্ছেনা।	ছোট কাহিনী বড় জঘন্য হচ্ছেনা	Anger	Anger
2532	এদেশে হত্যার বিচার চাইতে মানুষ লজ্জা পায় সে দেশে কপালে টিপ দেওয়ার প্রতিবাদ কিভাবে হয়।	এদেশে হত্যার বিচার চাইতে মানুষ লজ্জা পায় দেশে কপালে টিপ প্রতিবাদ কিভাবে	Joy	Joy
2533	আমার মতে পুলিশ ভাই কোন খারাপ কাজ করেনি। সে খুব ভালো একটি কাজ করেছে।	মতে পুলিশ ভাই খারাপ করেনি ভালো	Joy	Joy
2534	আমি বাংলাদেশি নাটকের নিয়মিত দর্শক। নাটক টা দারুন নিশো মানেই নতুন কিছু।কমিডি পাট টা অসাধারন অনেক হাসলাম।	বাংলাদেশি নাটকের নিয়মিত দর্শক নাটক দারুন নিশো মানেই কমিডি পাট অসাধারন হাসলাম	Joy	Joy
2535	নতুন বছরে একটা সুন্দর উপহার পেলাম খুব ভালো লাগলো।	বছরে একটা সুন্দর উপহার পেলাম ভালো লাগলো	Joy	Joy

Figure 3-5: Labeled Data

After preprocessing, validation, and labeling the three emotion classes of Joy, Sad and Anger, we got 1426 data for sad, 1331 data for anger and 1194 data for joy emotion class, 3951 textual data in total.

3.1.4 Inter–Annotator Reliability check

Based on the contexts, annotators can label one text as different emotion classes. The understanding, the mutual agreement between annotators is essential for labeling the data properly. So assessment of the agreement between the annotators is needed which gives a realistic view of the data quality. So we measured the inter –annotator reliability check using Cohen's kappa technique [10].

Cohen's Kappa (k): Cohen's kappa coefficient, $\mathbf{k}$ is a statistic that is used to measure inter-rater reliability and intra-rater reliability for categorical items. This k means how well two annotators agree with each other [14][15]. It is a metric often used to assess the agreement between them. The formula is

$$\boldsymbol{K} = {}^{\boldsymbol{Po} - \boldsymbol{Pe}}/_{\mathbf{1} - \boldsymbol{Pe}} \qquad (3\text{-}1)$$

Here,

Po = Relative observed agreement among annotators.

Pe = hypothetical probability of chance agreement.

If k = 1, the annotators are in complete agreement.

If k ≤ 0, the annotators disagree.

We measured Cohen's Kappa score of 96.3% which indicates the quality of the dataset.

3.1.5 Train Test split

The dataset is split in train and test using train_test_split function from scikit learn library where random_state is set to 200. The 80- 20 model is applied for splitting, where 80% data is used for training each model and the rest is for testing accuracy.

3.2 Feature Extraction

In order to train the classifiers, feature extraction is essential. It is a technique which converts textual data into numerical representation in a vector form. TF-IDF (Term frequency- Inverse document frequency) and word n -grams techniques are applied for feature extraction.

Table 3.1 represents the total number of data, number of words and number of types of words.

Table 3-1: Statistics of dataset

Category	No. of Data	Total No. of words	Total Types of Word
Joy	1194	8765	3845
Anger	1331	10224	4686
Sad	1426	10849	4823
Total	3951	29838	13354

3.2.1 Word n-gram

It is considered very functional and beneficial for classifying data. It is widely used in data analysis, NLP (Natural Language Processing), speech recognition, machine translation and so on [16].

Word n-gram is a group of successive terms in a text that can include word, numbers, symbols and punctuations. It is very effective for classifying data [7]. n –gram means a sequence of n number of words or terms.

Depending on the value of n, this process can be classified into following types:

- Unigram ; n = 1
- Bigram ; n = 2
- Trigram ; n = 3

Table 3.1 shows unigram, bigram, trigram in texts.

Table 3-2: Feature n –grams

বিদ্রোহ ছিলোনা ছিলো পরিকল্পিত হত্যা

Applying Word n-gram Feature

Uni-gram	**Bi-gram**	**Tri-gram**
বিদ্রোহ	বিদ্রোহ ছিলোনা	বিদ্রোহ ছিলোনা ছিলো
ছিলোনা	ছিলোনা ছিলো	ছিলোনা ছিলো পরিকল্পিত
ছিলো	ছিলো পরিকল্পিত	ছিলো পরিকল্পিত হত্যা
পরিকল্পিত	পরিকল্পিত হত্যা	
হত্যা		

3.2.2 TF-IDF

TF-IDF (Term frequency- inverse document frequency) is a statistical measure that informs about the relevancy of a word in a document [17]. It is measured by multiplying two matrices; the repetition of a word in a document (Term frequency) and the inverse document frequency, significance of the word across the document. “ *tf-idf*” statistic used [7]:

For a term *i* in document *j*:

$$\boldsymbol{tf - idf_{(i,j)} = tf_{i,j} * log(\frac{N}{df_i})} \qquad (3\text{-}2)$$

tf = Number of occurrences of i in j

df = Number of documents containing i

N= Total number of documents

Two dimensional feature matrices represent feature matrix where rows represent the reviews or sentences and columns represent unique words in the sentences of the given corpus. Each value in the matrix represents the *'tf-idf'* numerical value of a word occurred in a particular review or sentence.

A tabular representation of feature matrix is given in the following Table 3.5

Table 3-3: : TF-IDF value for textual data

Word No.	**Text No.**	**TF-IDF Value**
0	1663	1.0
1	48	0.49580894915310586
1	2140	0.6625190554779433
1	2004	0.5614641458439771
2	1762	0.6143330655024986
2	189	0.7890468203030179
3	525	0.7144200428246836
3	1204	0.6997170874077446

3.3 Classifiers & Implementation

Classification techniques are an essential part of NLP. Approximately 70% of the problems in NLP are classification problems. There are many classification models in machine learning. In this study, the used models are the following:

3.3.1 Support Vector Classifier (SVC)

Support vector machine is a machine learning algorithm that gives precise decision boundaries between vectors that belong in a group and those which don't belong in it. It is a standard classification and regression method. It is a discriminative classifier defined by a separating hyperplane. It draws a precise line or hyperplane in two-dimensional space dividing the space into two subspaces; for the belonging vectors of a group or category and for those which don't belong to it [18].

We worked with a support vector system which is another implementation for a support vector classifier that provides the method 'decision_function'. It provides confidence scores for the sample data. The objective of using SVC is to fit the provided data and return the best decision or hyperplane that classifies the data. The value is set to default parameter 1 with kernel 'rbf'.

3.3.2 Logistic Regression (LR)

Logistic regression is a supervised classification method that serves to solve a binary classification problem. The result is usually defined as 0 or 1 in the models of a double situation since the outcome is a probability. It estimates the probability of a given situation based on independent variables or a given corpus. It predicts a categorical variable which can be true or false, 0 or 1 and so on [19].

The linear function is applied to another function in the following relation:

$$\boldsymbol{(x) = g(\theta T x)} \qquad \textbf{(3-3)}$$

Where, $0 \leq h\theta \leq 1$

g represents the sigmoid function which can be expressed as follows:

$$\boldsymbol{(z) = 11 + e^{-z}} \qquad \textbf{(3-4)}$$

$$\boldsymbol{z = \theta t x} \qquad \textbf{(3-5)}$$

The random_state value is set to zero.

3.3.3 Decision Tree (DT)

Decision tree is a supervised machine learning algorithm. It evaluates basic decision tree rules derived from the features of the data and predicts the

target attribute [20]. It represents a tree-like structure as a classifier where the features of the given corpus are represented by internal nodes, decision rules represented by the branches and the outcome represented as each leaf node. The logic of this ML learning method is easier to understand because of its tree-like structure. It also follows or mimics the human thinking ability to giving a decision. For the parameter, random_state is set to 0.

Decision tree classifier is great for its interpretation and simplicity. It can be used for multidimensional data. Its tree-structured logic is easier to comprehend. The classification and learning sections are rapid and straightforward. Decision tree method is applied in various fields in real life such as manufacturing and production, civil planning, AI (Artificial Intelligence), astronomy, molecular biology and so on.

3.3.4 Multinomial Naive Bayes Classifier (MNB)

MNB is a probabilistic machine learning algorithm that is gaining popularity for text data classification in NLP. It is formed on Bayes theorem which predicts the tag of a text then calculates the probability for each tag or a given sample and then gives the tag with the highest probability as output [21].

The technique is to find out the probabilities of the emotion classes that are assigned to the textual data using the joint probabilities of the words and the classes. Bayes theorem is as follows:

$$\boldsymbol{P(A|B) = \frac{P(B|A)P(A)}{P(B)}} \quad \textbf{(3-6)}$$

Here,

A, B = two occurrences.

P(A), P(B) = probabilities of A & B

P(A|B) = probability of A given B is true.

P(B|A) = probability of B given A is true.

It takes the features of the data and takes the adjectives into account for each class. Then it gives the probability for that class.As parameter, we set α (additive smoothing parameter) to default value which is 1.

3.3.5 Random Forest Classifier (RF)

Random forest is an ensemble model using bagging as the ensemble method and decision tree as the individual model [22]. It is widely used for classification and regression problems. It uses decision trees on given data and classifies them by majority voting. It performs better for classification problems. It takes n number of random records from k number of records of the given dataset. For each sample, it constructs individual decision trees. For classification, the final output is considered after majority voting as each tree generates an output and voting is performed for every generated output.

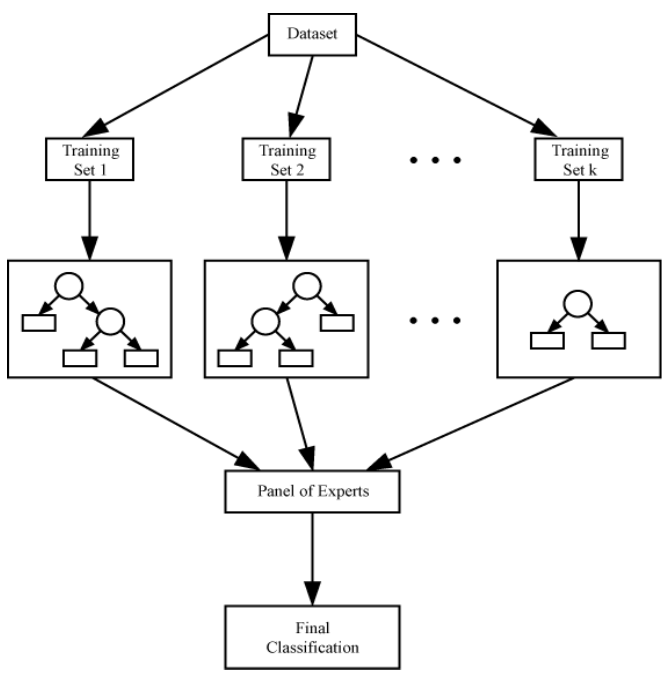

Figure 3-6: Figure Bagging classifier technique

For creating multiple decision trees, it does not use the same data for each decision tree. So it becomes less sensitive to the original training data. Different scoring methodologies can be applied to assess the features and create a root corresponding to that feature with the greatest influence on the label.

Scoring methodology like Gini index is used to decide the nodes on a tree branch.

$$Gini = 1 - \sum_{i=1}^{c} (Pi)2 \qquad \textbf{(3-7)}$$

Pi **=** relative frequency of the observed class

c = number of classes in the dataset

This formula uses the class and probability to measure the Gini of each branch on a node determining the possibility of the occurrence of the branches. For the Random Forest classifier, '*n_estimators'* is set to 200 and random_state is set to 0. *'n_estimators'* refers to the number of decision trees created before voting and choosing the final output.

3.3.6 Voting Classifier

In few types of ensemble technique, voting is the common one.
A voting classifier is a type of machine learning estimator that develops a number of base models or estimators and makes predictions based on averaging their results.Voting for each estimator output can be integrated with the aggregating criteria [23].

The voting criteria can be of two types [24]:

Hard Voting: Voting is calculated on the predicted output class.

Soft Voting: Voting is calculated on the predicted probability of the output class.

Voting criteria can be seen in figure 3.7

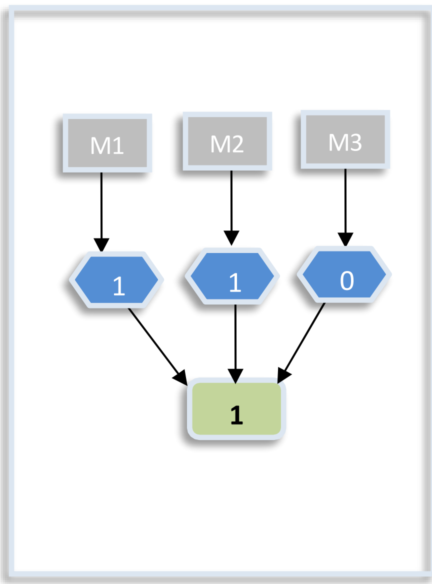

Figure 3-7: Hard Voting

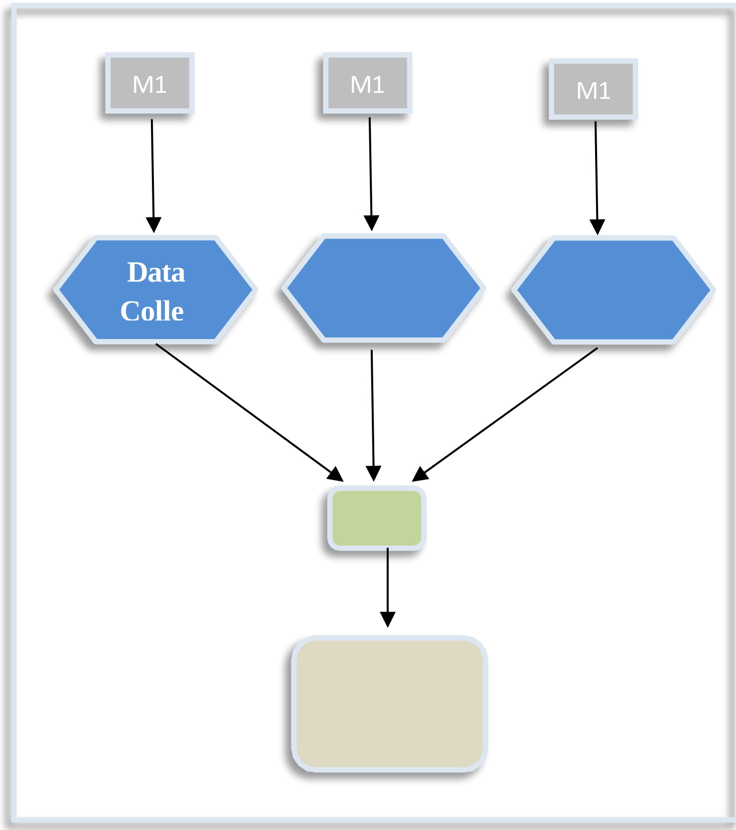

Figure 3-8: Soft Voting

For voting, there need to pass few base estimators.

So we pass three base estimators which are Multinomial Naïve Bayes, Support Vector classifier, Logistic Regression and we chose the voting criteria ‘soft’. Here we also set random_state to 0.

Chapter 4

Experimental Results Analysis & Discussion

This chapter presents the evaluation and analysis of the outcome of various ML models to categorize emotion classes from Bengali textual data.

To evaluate the performance of each model, precision, recall and F1-score for each emotion class are considered. To compare performance of each model, we considered accuracy.

4.1 Comparing Performances for ML Models

For unigram, SVC, MNB and VCM performed well than other classifiers as MNB reached highest accuracy with 67.383%, VCM with 67.25% accuracy and SVC with 66.371% accuracy. DT has the lowest accuracy score (56.76%).

Table 4.1 represents the performance estimation for unigram for all the ML models.

Table 4-1: For unigram-performance of ML models

Model Name	**Precision (%)**			**Recall (%)**			**F1 score (%)**			**Accuracy (%)**
	Anger	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	
SVC	63.059	82.587	59.006	61.0108	70.042	68.592	62.018	75.799	63.439	66.371
LR	62.2302	78.125	60.207	62.4548	73.8396	62.8158	62.342	75.921	61.484	65.992
MNB	68.4647	76.5957	59.682	59.5667	75.9493	67.87	**63.7056**	**76.2711**	63.5135	**67.383**
DT	58.297	65.6521	50.9202	49.4584	63.713	59.9277	53.515	64.668	55.058	56.76
RF	64.069	72.881	56.1728	53.4296	72.5738	65.7039	58.267	72.727	60.565	61.82
VCM	64.31	77.97	61.69	62.45	74.6835	65.703	63.39	76.26	**63.63**	**67.25**

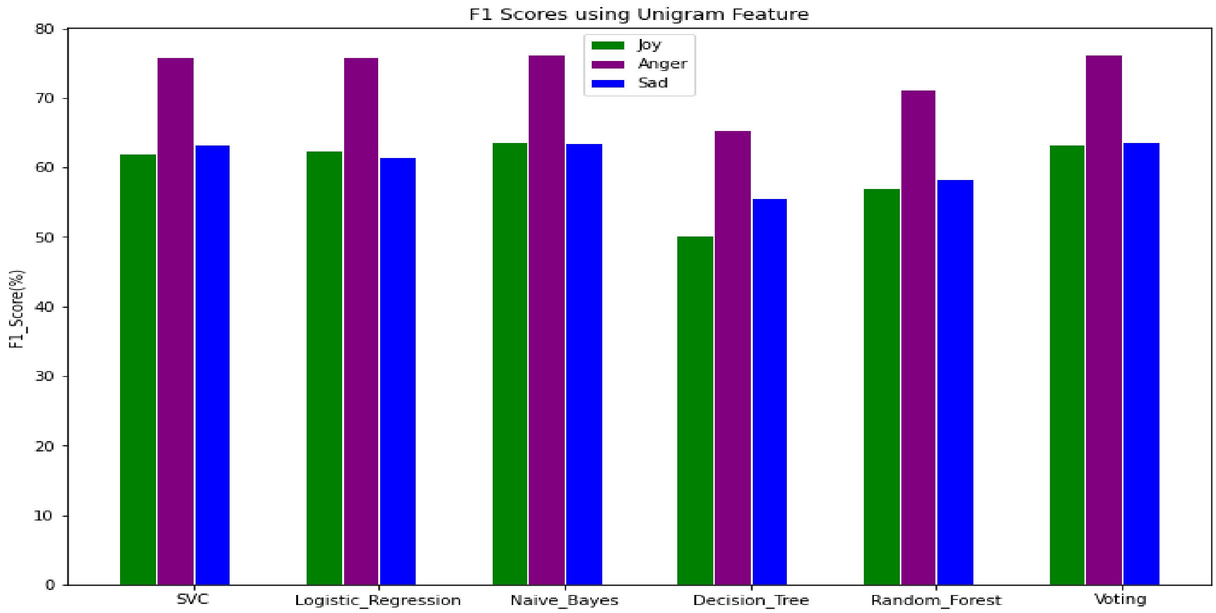

Figure 4-1: Graphical Representation of F-1 score

For bigram, we get the performance of each ML models where VCM reached highest accuracy with 67.509% where LR achieved 66.498% accuracy and MNB achieved 66.624% accuracy. DT has the lowest accuracy score with 54.48%. Table 4.2 represents the performance estimation for bigram for all the ML models.

Table 4-2: For bigram-performance of ML models

Model Name	**Precision (%)**			**Recall (%)**			**F1 score (%)**			**Accuracy (%)**
	Anger	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	
SVC	63.2575	83.248	57.272	60.288	69.198	68.231	61.737	75.576	62.273	65.739
LR	63.0996	79.545	60.00	61.7328	73.8396	64.9419	62.408	**76.586**	62.391	**66.498**
MNB	67.489	77.2925	58.307	59.205	74.683	67.148	63.076	75.965	62.416	**66.624**
DT	54.966	63.1025	47.3058	44.238	64.2911	56.0397	48.899	64.016	51.571	54.48
RF	61.538	70.638	55.59	51.9855	70.0421	64.6209	56.36	70.338	59.7662	61.56
VCM	65.648	78.571	60.983	62.093	74.261	67.148	**63.82**	76.35	**63.917**	**67.509**

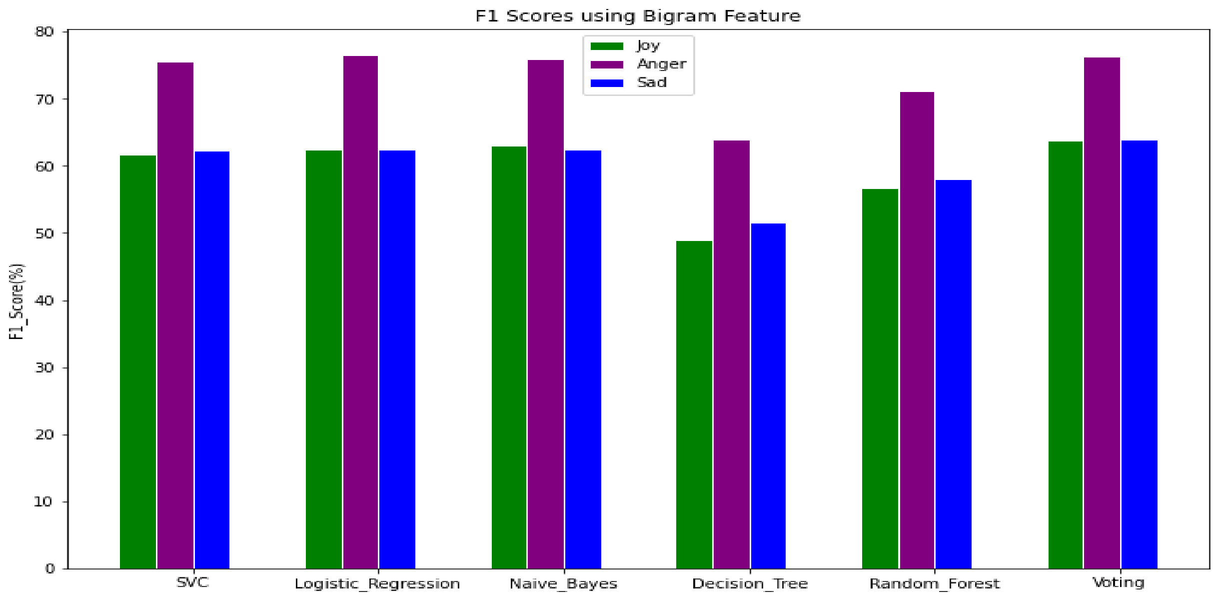

Figure 4-2: Graphical Representation of F-1 score

For trigram, VCM reached the highest accuracy with 67.509% score where MNB and LR were also able to achieve more than 66% accuracy (MNB with 66.498% accuracy & LR with 66.624% accuracy). DT gained the lowest score of accuracy which is 55.12%. Table 4.3 represents the performance estimation for trigram for all the ML models.

Table 4-3: : For trigram-performance of ML models

Model Name	**Precision (%)**			**Recall (%)**			**F1 score (%)**			**Accuracy (%)**
	Anger	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	**Anger**	**Joy**	**Sad**	
SVC	62.781	83.248	57.621	60.288	69.198	68.231	61.5101	75.576	62.4793	65.739
LR	63.3333	79.5454	61.132	61.732	73.839	65.3429	62.522	**76.5864**	62.629	**66.624**
MNB	67.3553	76.9565	58.307	58.844	74.6835	67.148	62.8131	75.8029	62.4161	**66.498**
DT	56.2211	60.474	50.1557	44.0433	64.5569	58.122	49.392	62.4489	53.846	55.12
RF	60.905	70.588	54.838	53.4296	70.886	61.371	56.923	70.736	57.921	62.199
VCM	65.648	78.571	60.983	62.093	74.261	67.148	**63.82**	76.35	**63.917**	**67.509**

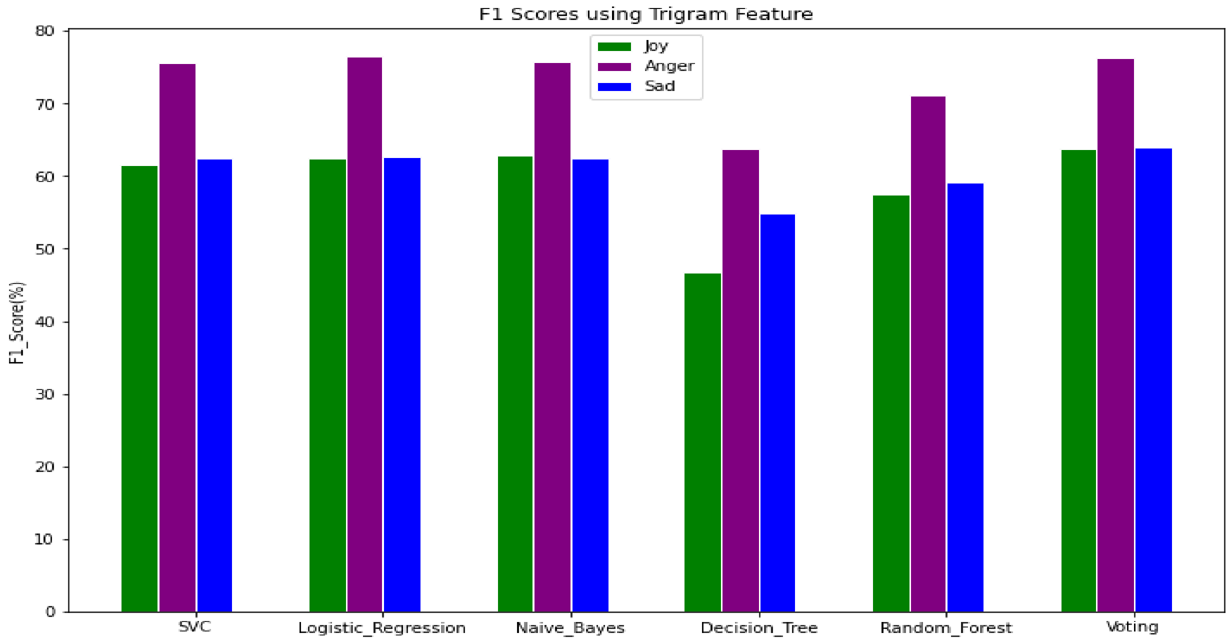

Figure 4-3: Graphical Representation of F-1 score

4.2 Learning Curve

The First graph shows the learning curve for classifiers for our emotion detection. We evaluated the model's error by splitting our dataset into a training and a validation set. Model estimation was done by taking one single instance from the training set and then we measured error of the model on the validation set and on that single training set.

The second plot shows the times required by the model to train with various sizes of training dataset.

The third plot shows how much time was required to train the models for each training size and how the score changed within that time frame.

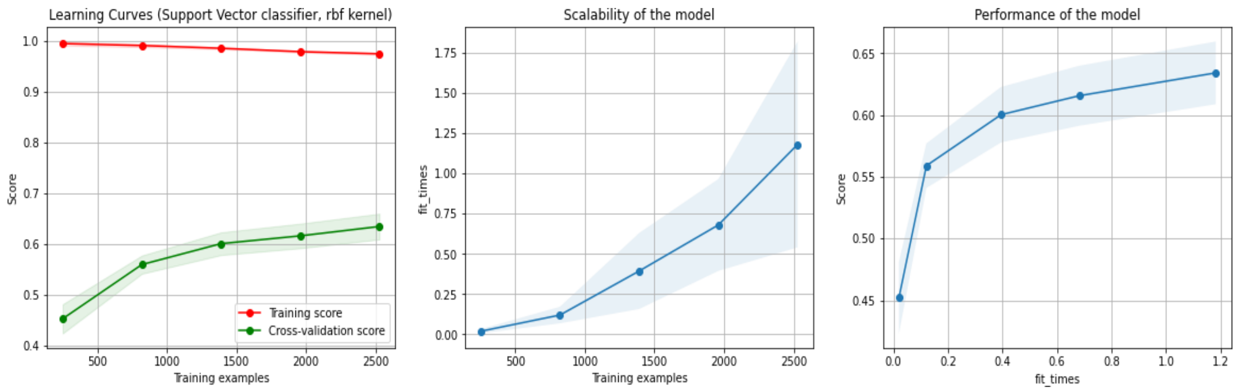

Figure 4-4: Learning Curves, Scalability and performance of SVC using Word-Uni gram

Figure 4-5: Learning Curves, Scalability and performance of LR using Word-Uni gram

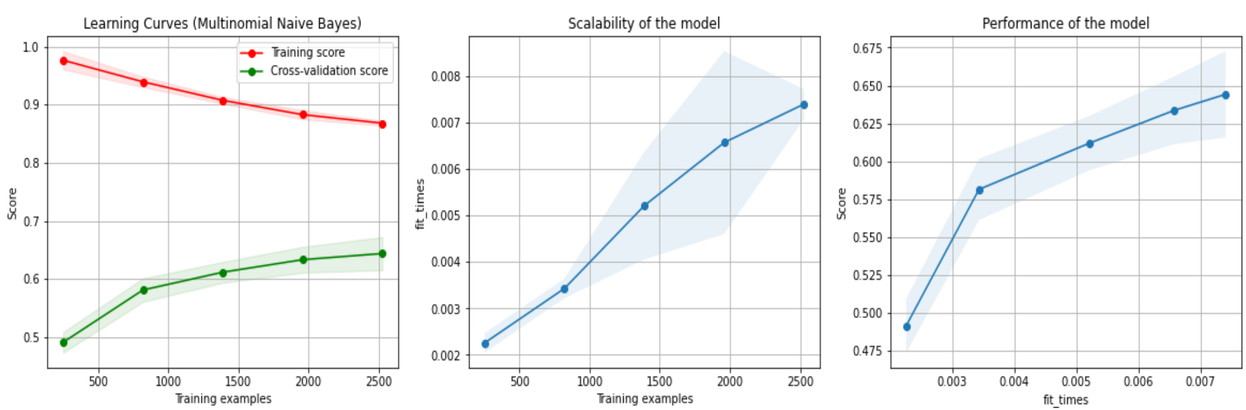

Figure 4-6: Learning Curves, Scalability and performance of MNB using Word-Uni gram

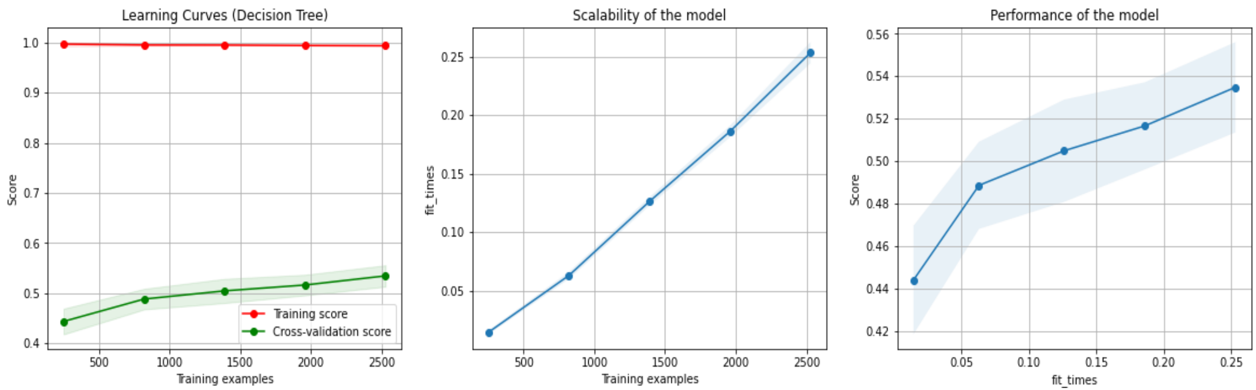

Figure 4-7: Learning Curves, Scalability and performance of DT using Word-Uni gram

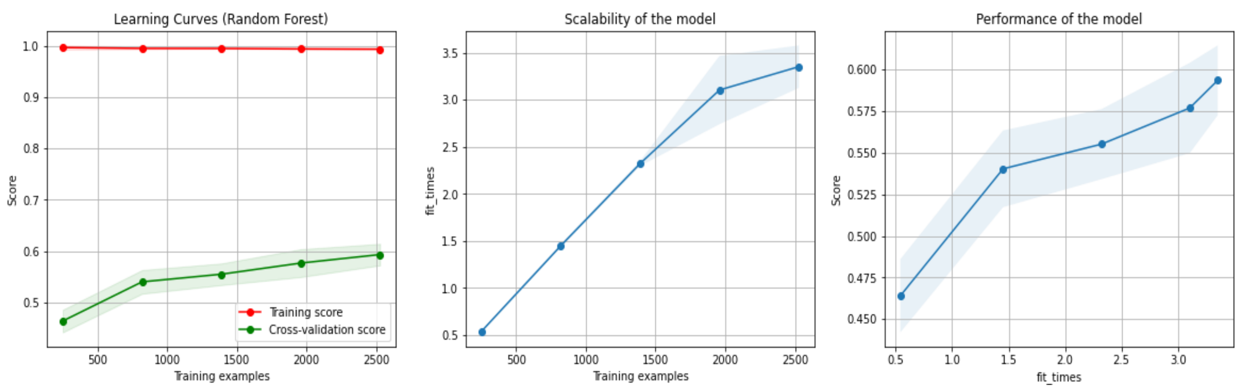

Figure 4-8: Learning Curves, Scalability and performance of RF using Word-Uni gram

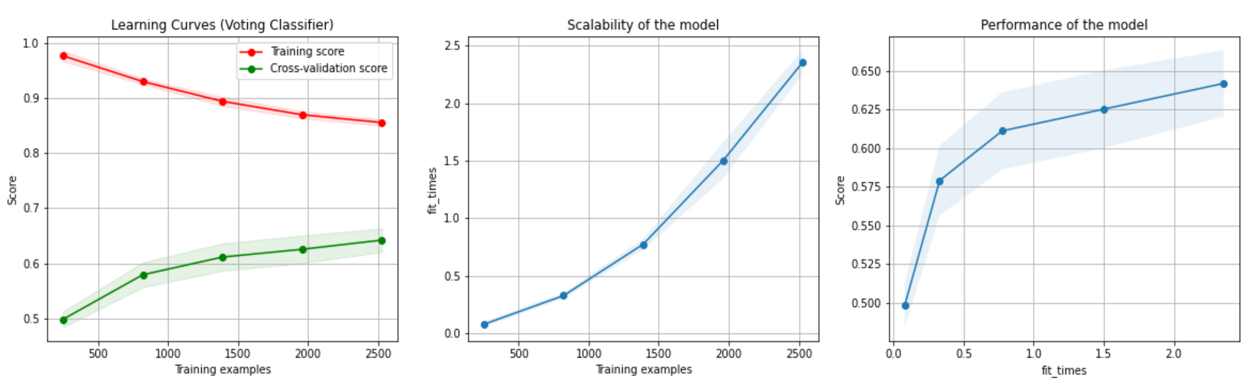

Figure 4-9: Learning Curves, Scalability and performance of VC using Word-Uni gram

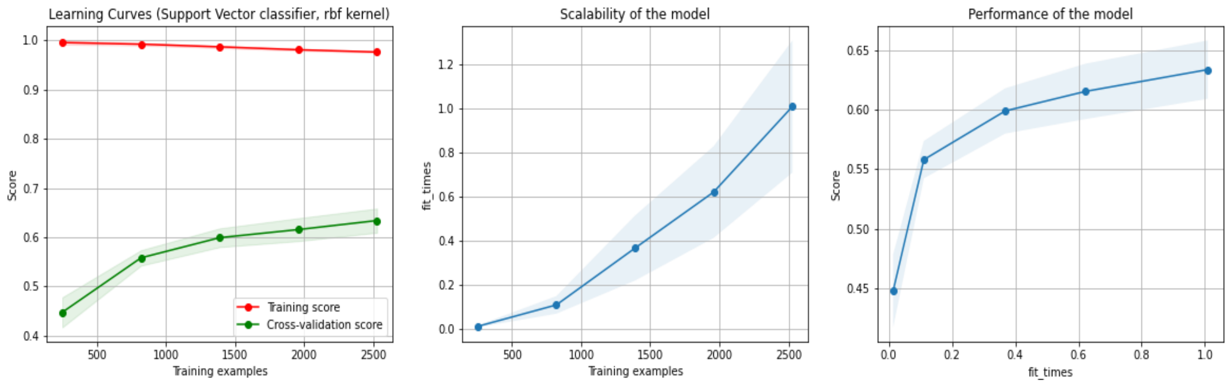

Figure 4-10: Learning Curves, Scalability and performance of SVC using Word-Bi gram

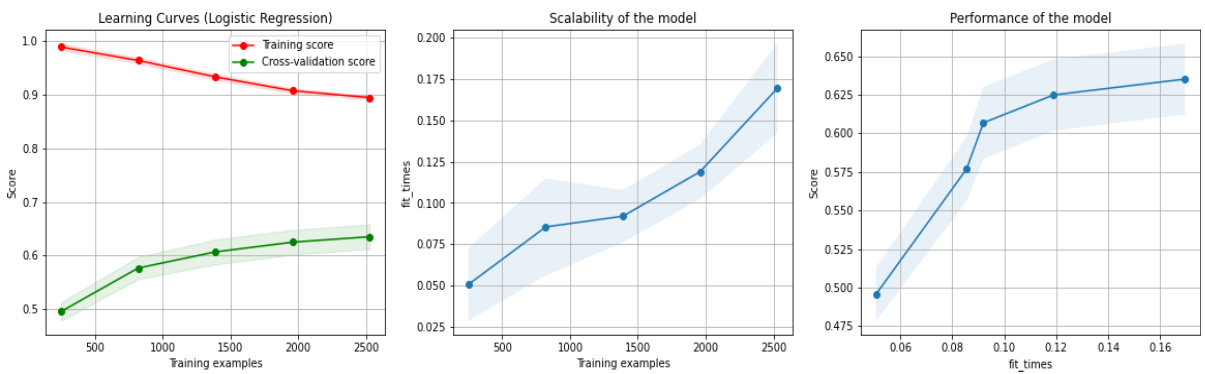

Figure 4-11: Learning Curves, Scalability and performance of LR using Word-Bi gram

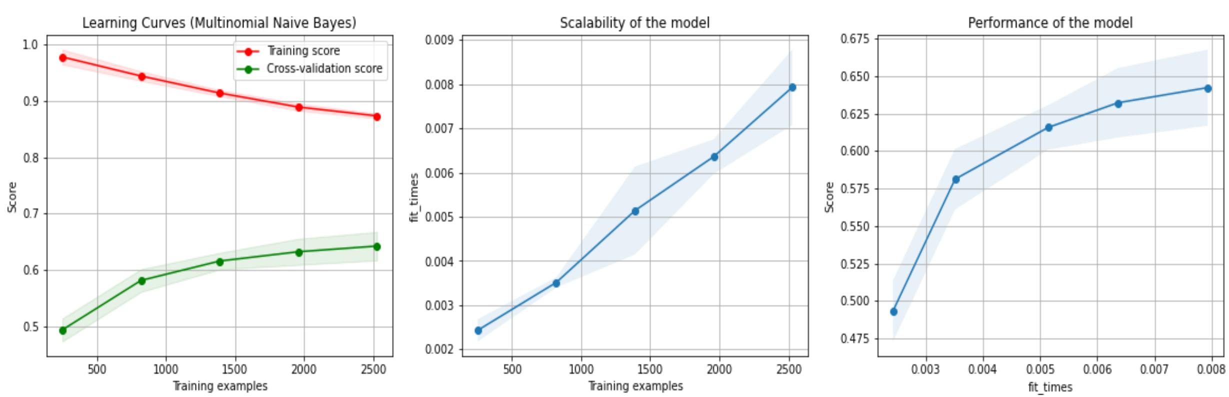

Figure 4-12: Learning Curves, Scalability and performance of MNB using Word-Bigram

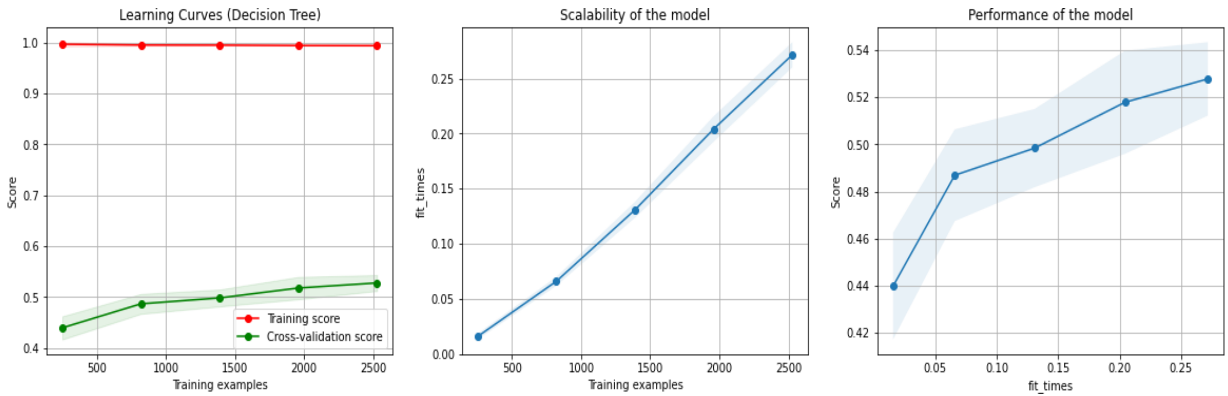

Figure 4-13: Learning Curves, Scalability and performance of DT using Word-Bi gram

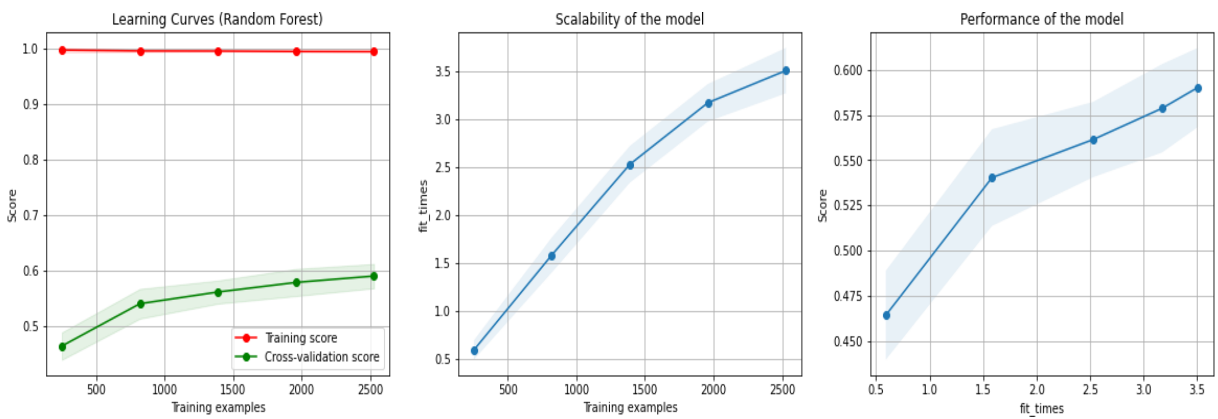

Figure 4-14: Learning Curves, Scalability and performance of RF using Word-Bi gram

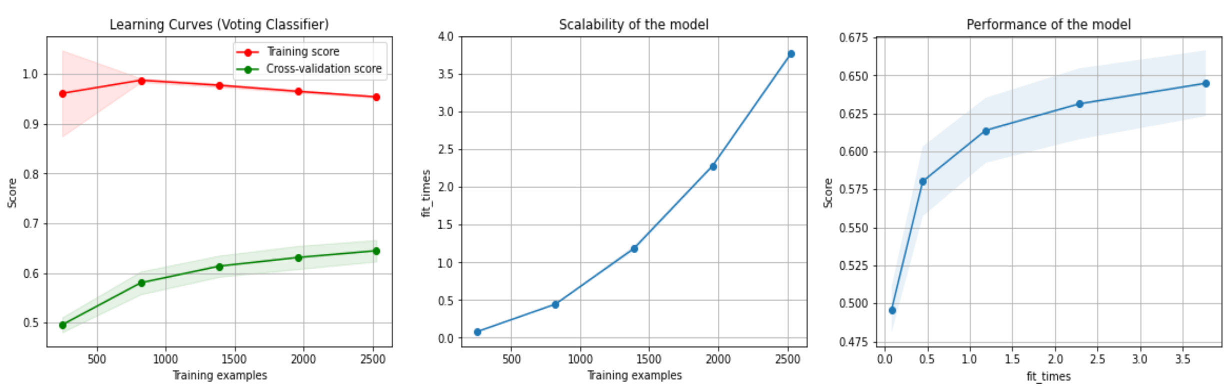

Figure 4-15: Learning Curves, Scalability and performance of VC using Word-Bi gram

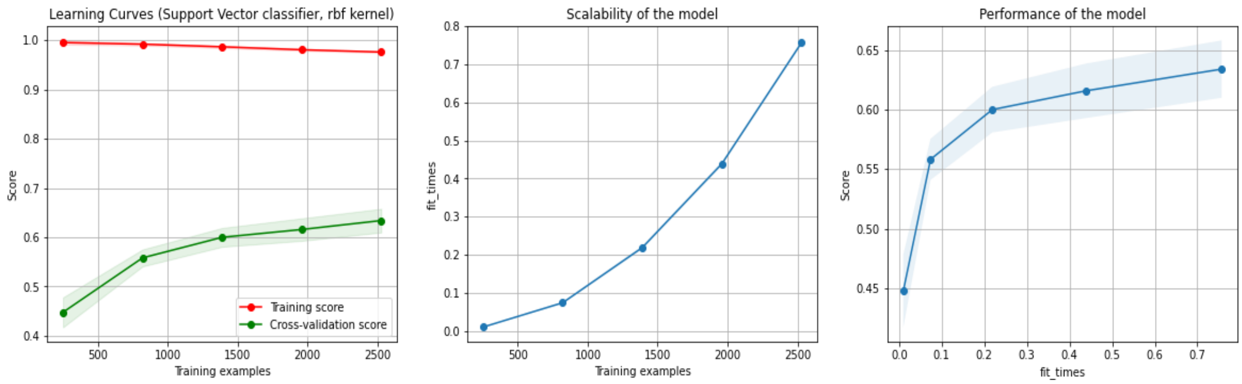

Figure 4-16: Learning Curves, Scalability and performance of SVC using Word-Tri gram

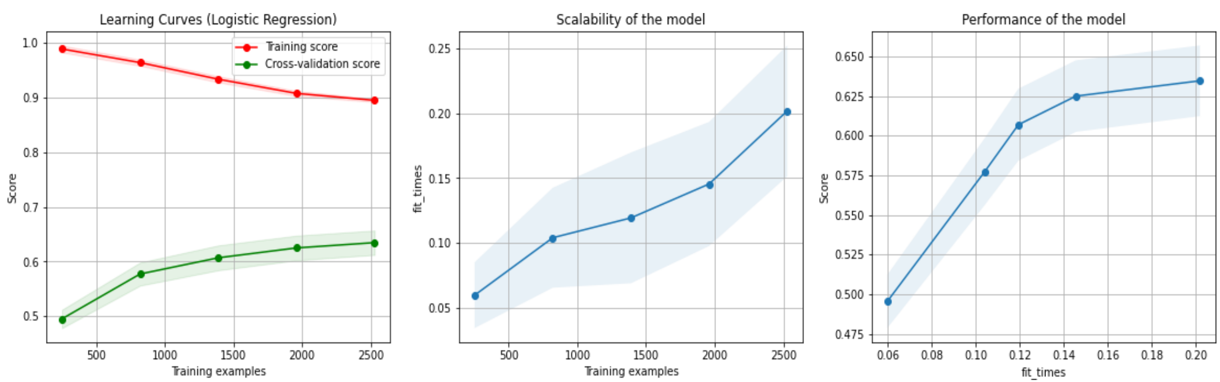

Figure 4-17: Learning Curves, Scalability and performance of LR using Word-Tri gram

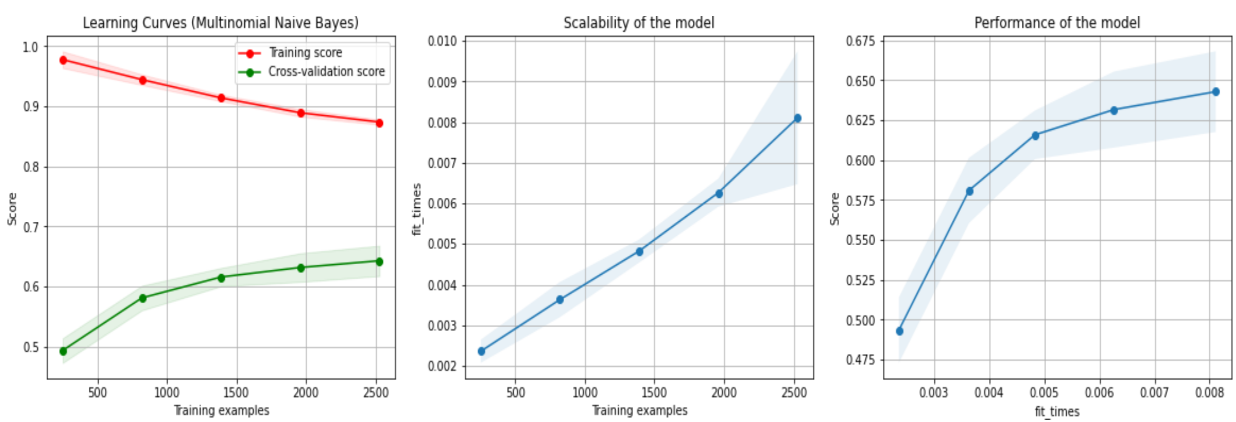

Figure 4-18: Learning Curves, Scalability and performance of MNB using Word-Tri gram

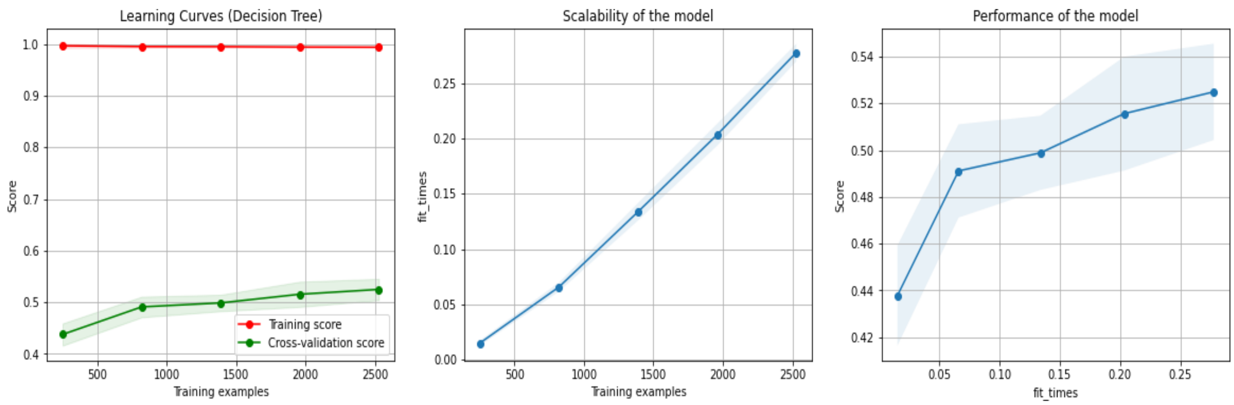

Figure 4-19: Learning Curves, Scalability and performance of DT using Word-Tri gram

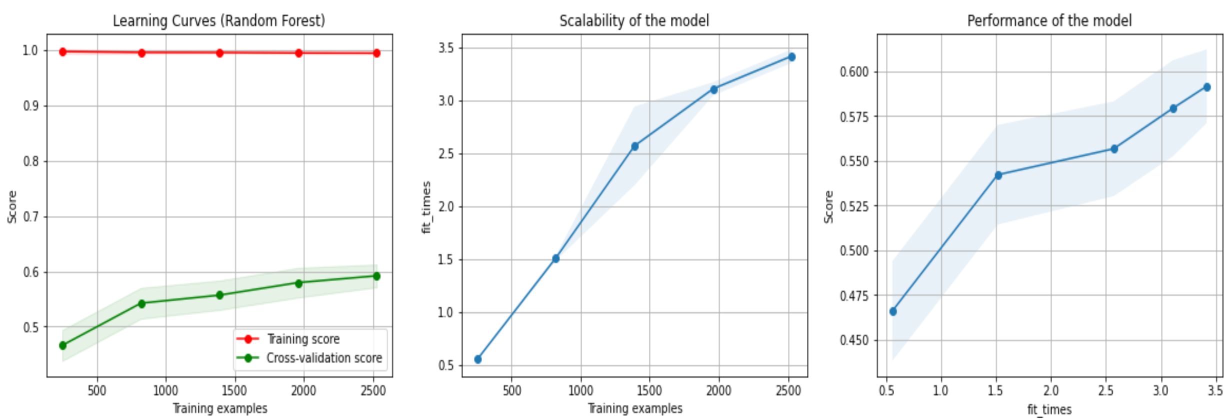

Figure 4-20: Learning Curves, Scalability and performance of RF using Word-Tri gram

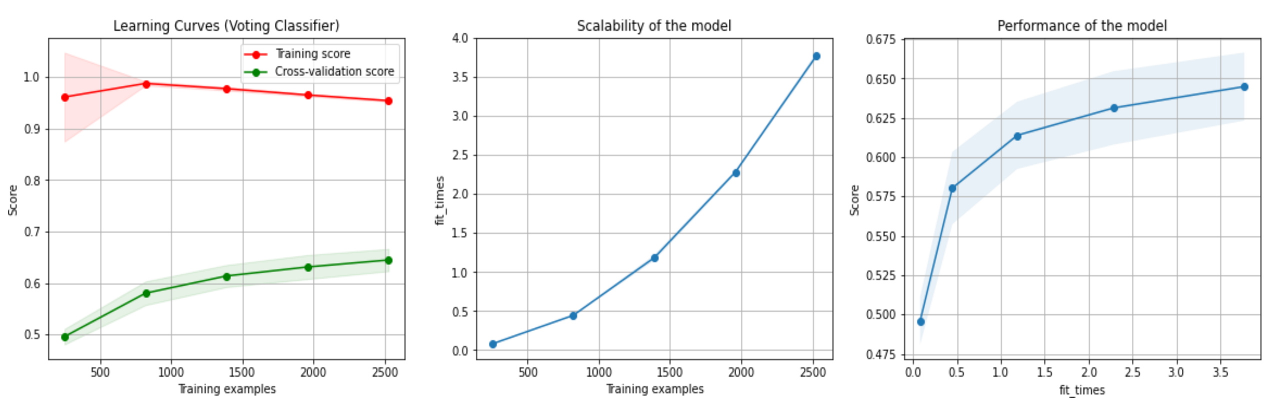

Figure 4-21: Learning Curves, Scalability and performance of VC using Word-Tri gram

4.3 Confusion Metrics

Confusion metrics breaks down the prediction results of a classification problem. It gives a view of a particular classifier making errors based on the problem. The figures below represent the error analysis of each ML classifiers using confusion metrics. Each row represents the correctly labeled classes. Each column represents the prediction of the classifier for each class. Confusion metrics are shown for unigram, bigram, trigram for each ML model.

Table 4-4: Confusion metrics (SVC- Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	169	18	90
	Joy	29	166	42
	Sad	70	17	190

Table 4-5: Confusion metrics (LR - Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	173	25	79
	Joy	26	175	36
	Sad	79	24	174

Table 4-6: Confusion metrics (MNB- Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	165	23	89
	Joy	19	180	38
	Sad	57	32	188

Table 4-7: Confusion metrics (DT - Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	137	38	102
	Joy	28	151	58
	Sad	70	41	166

Table 4-8: Confusion metrics (RF - Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	148	36	93
	Joy	19	172	49
	Sad	67	28	182

Table 4-9: Confusion metrics (VCM - Unigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	169	25	83
	Joy	24	176	37
	Sad	67	26	184

Table 4-10: Confusion metrics (SVC - Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	167	16	94
	Joy	26	164	47
	Sad	71	17	189

Table 4-11: Confusion metrics (LR – Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	167	24	82
	Joy	24	175	38
	Sad	76	21	180

Table 4-12: Confusion metrics (MNB - Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	164	21	92
	Joy	19	177	41
	Sad	60	31	186

Table 4-13: Confusion metrics (DT – Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	117	42	118
	Joy	29	150	58
	Sad	77	42	158

Table 4-14: Confusion metrics (RF - Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	144	37	96
	Joy	24	166	47
	Sad	66	32	179

Table 4-15: Confusion metrics (VCM – Bigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	165	23	89
	Joy	22	177	38
	Sad	64	25	188

Table 4-16: Confusion metrics (SVC - Trigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	167	16	94
	Joy	28	164	45
	Sad	71	17	189

Table 4-17: Confusion metrics (LR – Trigram)

		Model's Prediction		
		Anger	**Joy**	**Sad**
Experts Label	**Anger**	171	24	82
	Joy	24	175	38
	Sad	75	21	181

Table 4-18: Confusion metrics (MNB - Trigram)

		Model's Prediction		
		Anger	Joy	Sad
Experts Label	Anger	163	22	92
	Joy	19	177	41
	Sad	60	31	186

Table 4-19: Confusion metrics (DT – Trigram)

		Model's Prediction		
		Anger	Joy	Sad
Experts Label	Anger	122	51	104
	Joy	28	153	56
	Sad	67	49	161

Table 4-20: Confusion metrics (RF - Trigram)

		Model's Prediction		
		Anger	Joy	Sad
Experts Label	Anger	148	36	93
	Joy	22	168	47
	Sad	73	34	170

Table 4-21: Confusion metrics (VCM – Trigram)

		Model's Prediction		
		Anger	Joy	Sad
Experts Label	Anger	165	23	89
	Joy	22	177	38
	Sad	65	25	187

The confusion metrics give the total view of the error analysis for each classifier. Predictions for each class of the classifiers from the actual labels are shown in the metrics. For example, in table 4.21, actual label is Anger and the VCM classifier's prediction is Anger, total number of this kind of texts are 165. Actual label is Joy but the model predicts as Anger, there are 22 texts. Actual label is Sad but the model predicts as Anger, there are 65 texts of this type.

Chapter 5

Conclusion

This work investigated the outcome and the performance of different ML (Machine Learning) techniques (SVC, LR, RF, DT, MNB, Voting) on the developed dataset for detecting emotion from Bengali textual data. Due to the scarcity of standard corpora, we built a dataset of Bengali texts that contains 3951 texts of three emotion classes: joy, sad, anger. We get Cohen's kappa score of 0.96 which reflects the quality of the dataset. We applied word n -gram, TF-IDF techniques for feature extraction.

From all the models, we see that VCM is superior and performed with highest score of accuracy (67.509% for both bigram & trigram) on the dataset. For unigram MNB reached highest with the accuracy score of 67.38%.

Some difficulties were faced as due to the scarcity of benchmark corpora on Bengali texts. Datasets on Bengali texts were very rare and available datasets were very noisy. The developed dataset had roughly 4500 data which was harvested from various online platforms using API. We only considered three emotion classes from six basic emotions: joy, sad, anger. Even after preprocessing and data labeling, there were lack of consistency in annotations and in removal of noises which makes the models being unable to reach higher accuracy. Also, overall estimation and assessment would have been much better if more data were collected.

Since we faced difficulties in developing the corpora and also in feature extraction, we want to work on collecting more data, better techniques of feature extraction, work with Deep Learning (DL) models and on the other emotion classes which are not included in this study.

References

[1] M. Soleymani, D. Garcia, B. Jou, B. Schuller, S.-F. Chang, and M. Pantic, “A survey of multimodal sentiment analysis,” *Image Vis. Comput.*, vol. 65, pp. 3–14, Sep. 2017, doi: 10.1016/j.imavis.2017.08.003.

[2] P. Ekman, “Facial expression and emotion.,” *Am. Psychol.*, vol. 48, no. 4, pp. 384–392, 1993, doi: 10.1037/0003-066X.48.4.384.

[3] S. Kusal, S. Patil, K. Kotecha, R. Aluvalu, and V. Varadarajan, “AI Based Emotion Detection for Textual Big Data: Techniques and Contribution,” *Big Data Cogn. Comput.*, vol. 5, no. 3, 2021, doi: 10.3390/bdcc5030043.

[4] M. Hasan, “Bengali Language,” 2022. https://archive.dhakatribune.com/world/2020/02/17/bengali-ranked-at-7th-among-100-most-spoken-languages-worldwide.

[5] S. Azmin and K. Dhar, “Emotion Detection from Bangla Text Corpus Using Naïve Bayes Classifier,” *2019 4th Int. Conf. Electr. Inf. Commun. Technol. EICT 2019*, no. December, pp. 1–5, 2019, doi: 10.1109/EICT48899.2019.9068797.

[6] A. Das, O. Sharif, M. M. Hoque, and I. H. Sarker, “Emotion Classification in a Resource Constrained Language Using Transformer-based Approach,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, 2021, pp. 150–158, doi: 10.18653/v1/2021.naacl-srw.19.

[7] T. Parvin, O. Sharif, and M. M. Hoque, “Multi-class Textual Emotion Categorization using Ensemble of Convolutional and Recurrent Neural Network,” *SN Comput. Sci.*, vol. 3, no. 1, 2022, doi: 10.1007/s42979-021-00913-0.

[8] F. F. Nádia, E. R. Hruschka, and E. R. H. Jr, “Tweet sentiment analysis with classi fi er ensembles,” *Decis. Support Syst.*, 2014, doi: 10.1016/j.dss.2014.07.003.

[9] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, “Sentiment classification: The contribution of ensemble learning,” *Decis. Support Syst.*, vol. 57, pp. 77–93, Jan. 2014, doi: 10.1016/j.dss.2013.08.002.

[10] M. A. Iqbal, A. Das, O. Sharif, M. M. Hoque, and I. H. Sarker, “BEmoC: A Corpus for Identifying Emotion in Bengali Texts,” *SN Comput. Sci.*, vol. 3, no. 2, pp. 1–17, 2022, doi: 10.1007/s42979-022-01028-w.

[11] taymaytomo, “Data Scraping.” 2022, [Online]. Available: https://github.com/taymaytomo/youtube/blob/main/Download or Scrape all YouTube Comments and Replies.gs.

[12] N. S. Dash and L. Ramamoorthy, *Utility and application of language corpora*. 2018.

[13] K. A. Avishek Das, “BnPreprocessing-2.” [Online]. Available:

https://pypi.org/project/BnPreprocessing/.

[14] Kurtis Pykes, "CK." https://towardsdatascience.com/cohens-kappa-9786ceceab58.

[15] M. Bland, "Cohen ' s Kappa Statistic," no. 1892, pp. 188–189, 2019.

[16] S. Srinidhi, "Word n-gram." https://towardsdatascience.com/understanding-word-n-grams-and-n-gram-probability-in-natural-language-processing-9d9eef0fa058.

[17] C. Sammut and G. I. Webb, Eds., "TF–IDF BT - Encyclopedia of Machine Learning," Boston, MA: Springer US, 2010, pp. 986–987.

[18] C. Zhang, X. Shao, and D. Li, "Knowledge-based Support Vector Classification Based on C-SVC," *Procedia Comput. Sci.*, vol. 17, pp. 1083–1090, Dec. 2013, doi: 10.1016/j.procs.2013.05.137.

[19] D. O. M. Thomas W. Edgar, "Research Methods for Cyber Security," 2017, [Online]. Available: https://www.sciencedirect.com/topics/computer-science/logistic-regression?fbclid=IwAR2VUBewfzUN3hoUSkUCLX1Ut9AcIbDKkX-JxyxJ5aJJKCvT5RksPr3qFxQ.

[20] J. Fürnkranz, "Decision Tree BT - Encyclopedia of Machine Learning," C. Sammut and G. I. Webb, Eds. Boston, MA: Springer US, 2010, pp. 263–267.

[21] Shriram, "Multinomial Naive Bayes Explained: Function, Advantages & Disadvantages, Applications in 2022," 2022. https://www.upgrad.com/blog/multinomial-naive-bayes-explained/.

[22] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, "RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12343 LNCS, 2020, pp. 503–515.

[23] A. Nair, "Voting classifier," *Towards Data Science*, 2021. https://towardsdatascience.com/combine-your-machine-learning-models-with-voting-fa1b42790d84.

[24] S. Kumar, "Use Voting Classifier to improve the performance of your ML model," *Towards Data Science*, 2021. https://towardsdatascience.com/use-voting-classifier-to-improve-the-performance-of-your-ml-model-805345f9de0e.