Bengali social media comments classification and toxicity detection using advanced machine learning algorithms

By
Rayhan Rafin
213-15-4278

Mohammad Sohaib Islam Shibly
213-15-4258

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements forthe **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by

Tapasy Rabeya
Senior Lecturer
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by

Mr. Showmick Guha Paul
Lecturer
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

September 16, 2025

APPROVAL

This Project titled **"Bengali social media comments classification and toxicity detection using advanced machine learning algorithms"**, submitted by **Rayhan Rafin (ID:213-15-4278)** and **Mohammad Sohaib Islam Shibly (ID: 213-15-4258)** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **16 September, 2025**.

BOARD OF EXAMINERS

Dr. Sheak Rashed Haider Noori (SRH) — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Sadekur Rahman (SR) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Saiful Islam (SI) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Arshad Ali (DAA) — **External Examiner**
Professor
Department of Computer Science and Engineering
Hajee Mohammad Danesh Science & Technology University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Tapasy Rabeya, Senior Lecturer**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this projectnor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Tapasy Rabeya (TRA)
Senior Lecturer
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:

Mr. Showmick Guha Paul (SGP)
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Rayhan

Rayhan Rafin
213-15-4278
Department of Computer Science and Engineering
Daffodil International University

Shibly

Mohammad Sohaib Islam Shibly
213-15-4258
Department of Computer Science and Engineering
Daffodil International University

ACKNOWLEDGEMENTS

This work may not have been possible without the support and contributions of many authors and supervisor over the past two semesters. We are grateful to everyone who assisted us in any way.

First, we would like to express our thanks and gratefulness to the almighty for His divine blessing making it possible for us in completing the **Final Year Design Project (FYDP)** successfully.

We are very grateful and wish our indebtedness to our supervisor, **Tapasy Rabeya**, **Senior Lecturer**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Natural Language Processing (NLP)** to carry out this project. Her patience, scholarly advice, perpetual energetic, constant guardianship, wise criticism, extensive advice and time devoted to reading often lesser drafts and correcting them at all productivity points have made this project possible.

We would also like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his cooperation in finishing our project and also to other faculty members and the staffs of the Department of Computer Science and Engineering, Daffodil International University.

We want to thank our entire course mates at Daffodil International University, who had a part in this discussion while completing the coursework.

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

ABSTRACT

This project presents a system for classifying topics and sentiments from Bengali comments using both machine learning and deep learning algorithms. Detecting toxic behavior and hate speech is a primary usage of this system. All steps taken to complete this experiment were done maintaining widely accepted standards. One of the challenges of this experiment was working with a low-resource language like Bengali that required custom and extensive preprocessing and model design. All the trained models were evaluated using standard metrics. Aside from the technical aspects, this project also focuses on improving online safety and moderation for Bengali language. A comment analysis tool for web view is also developed as part of this project.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

With the rise of social media comes a huge wave of data and a form of that is social media comments. With the comments comes the task of moderating, analyzing, censoring to make it a safe place for everyone. There has been a rise of cyber bullying and toxic comments which leads to suicide. Artificial Intelligence may be able to automate the task of moderation and censorship and aid site owners as well as analysts.

As the internet grows so did its data and countermeasures for the data moderation. However, most of them are fitted to handle English language. Regional language moderation like Bengali language is still far from perfect. With new data and advanced models, we can improve this insufficiency. This research aims to solve it by using advanced Neural network algorithm like CNN,LSTM as well as classical machine learning models.

1.2 Motivation

The motivation of our research comes from the demand for better models and linguistic challenges [14] posed by Bengali language. Unlike English Bengali language have non standard spelling, inconsistent grammar and lack of structural uniformity leads to poor model generalization. Mitigating these challenges needs better data, preprocessing, sampling and model architecture.

By improving toxic content detection and classification by topic we aim to give analysts a chance to detect trends and for social media owners to improve moderation in their site. This will result in a better social media for everyone and improve research quality by analysts on trends.

1.3 Objectives

The purpose of this research is to reach specific objectives that will make the project worthwhile. The key objectives are:

i. The main goal of this research is to identify toxic and censorable comments. This includes abusive and threat from topic label and hate from sentiment label.
ii. Our secondary goal lies in classifying comments by topic so that analysts can find them by category and do sentiment evaluation on them.
iii. Using advanced algorithm we intend to find out sentiment of comments (positive,

negative, neutral, hate), helping identify public reaction.

1.4 Methodology

The project initially created a systematic way of collecting and preprocessing Bengali comments, using approaches like stemming, tokenization, text cleaning etc. followed by creating a clean dataset. After the dataset was prepared, we implemented a wide variety of models, including more classic machine learning models and deep learning models which were also combined to make more hybrid models. Each of these models were trained and tuned in Google Colab, and we used stratified splits to keep some balance in the data. Finally, we then evaluated each of the models using accuracy, confusion matrix, ROC curve and precision recall curve to compare performance of the models for topic, sentiment and possible toxicity classification.

1.5 Project Outcome

One of the outcomes of this research is aiding other research by contributing the primary data [3] we collected. This research may also open doors to new ideas on model building for under-represented classes. It will help in toxic comments detection and may become the foundation of an automated moderation tool for Bengali language. It can also be implemented in browser extension for child safe mode and content blocking.

The topic classification and sentiment prediction on those specific topics will also help analysts to understand public reaction to events and news. Overral, this project will help push advancement in Bengali NLP research.

1.6 Organization of the Report

Chapter 1 (Introduction) describes the overview of the project by presenting background, aims, methodology, expected outcomes and providing a brief outline of the report structure.

Chapter 2 (Background) provides the required theoretical and practical knowledge necessary to understand the project, the necessary literature review and description of relevant existing system as well as a gap analysis to note any opportunities for future research.

Chapter 3 (Research Methodology) describes the method in detail including requirement analysis, system design, project scheduling, role assignments and an overall description of the process for model development.

Chapter 4 (Implementation and Results) covers the configuration of the development environment, implementation of models, description of testing and evaluation processes, performance criteria, comparison of other models and discussion of the findings.

Chapter 5 (Engineering Standards and Design Challenges) covers where such systems, models and applications align with the associated software, hardware and communication standards. Also, the social and environmental implications, any social or ethical concerns, sustainability, project management, financial evaluation or budget analysis and management of complex engineering projects.

Chapter 6 (Conclusion) summarizes the key findings, mentions any limitations and where the project could go in the future.

Chapter 2

Background

2.1 Introduction

The expansion of social media and online platforms produces considerable amounts of Bengali comments every day. These sources provide various textual data with opinions, sentiments, or sometimes toxic content. There are many steps before classifying text. These documents need to be processed using Natural Language Processing techniques and they need to pass through a variety of preprocessing steps such as tokenization and stemming. These steps will be important in the case of Bengali because it is a complex, under-resourced language.

Machine learning behavior classifiers and Deep learning models can be useful for classification tasks such as topic and sentiment classification. Recurrent learning methods can combine the benefits of both machine learning and deep learning. In this work, we trained models and then tested the models across three representative Bengali comment datasets assessing the comments with topic and sentiment classifiers as a build-future capability to detect toxicity.

2.2 Literature Review

Amit et al.[1] developed a hate speech detection system from Bengali social media. From a dataset of 7425 comments and seven hate class the models were built. It involves preprocessing like stemming, stop word removal, custom emoticon module to capture emojis. TF-IDF and word embedding were used to extract feature. Classification was done by encoder decoder where cnn was the encoder and RNN based models as decoder. Their attention based model reached highest performance with 77% accuracy

Another approach was done by Reazul et al [2] by building a model that not only classify hate speech but also explain why the decisions were made. Trained on a ensemble of transformer based models like Bangla BERT and XLM-RoBERTa it is able to detect four types of hate. They used sensitivity analysis and layer wise relevance propagation to highlight important words. Their approach reached 88% f1 score in general. They also contributes labelled dataset and interpretation tools for it.

A hybrid of CNN and LSTM called CLSTM was proposed by Rezaul et al [4]. They took a large number of dataset about 42,036 comments and applied preprocessing steps on it. The study was done on six machine learning models and 4 deep learning model. Among them the hybrid CLSTM was able to capture both local and sequential features of comments. It achieved an accuracy of 85.8% and 0.86 f1 score. A website

was also built for practical usage of the model

Nauros et al. [5] annotated 30,000 comments from crowdsourcing under certain criteria. They used SVM, LSTM and BiLSTM with Word2Vec, FastText and BengFastText . among the models SVM outperformed other deep neural network with 87.5% accuracy.
A different approach was done by Abdul et al [6] where they translated the Bengali text into English before further training. Both direct and dictionary based translation was done using google translate. After pre processing a BOW model was used for feature extraction and Naive Bayes for classification. The model achieved results using 10 fold cross validation. This gives opens a new path of exploration in Bangla comment analysis.

Nayan et al [7] explored how supervised models can perform on hate detection. Using a public dataset of 10,219 comments they compared machine learning models with deep learning models. They found that deep learning models significantly outperform ML models. One of the model CNN was able to get an accuracy of 95.3%. This shows how neural network can capture local patterns in Bengali informal text which often contains phrases.

In their research Puja et al [8] used a balanced dataset of 5600 comments from facebook. They explored both traditional models like Multinomial Naïve Bayes, SVM and a hybrid CNN-LSTM model. In their research they found Naïve Bayes underperforming while linear SVM gave 78% accuracy. As the dataset grew their hybrid model performance jumped from 65.5% accuracy to 77.5%. This indicates usefulness of deep learning model on large dataset.

Like Puja Estiak et al [9] explored both traditional and hybrid deep learning models. They trained models on 4700 comments categorized into seven labels. They introduced Bengali specific stemming rules based on grammar which improved their accuracy. Among all models RNN-LSTM performed the best with 82.2% accuracy. Their study found ANN lagging behind other models. Multiple feature extraction techniques like CountVectorizer, TF-IDF, word embedding were used. Their study goes to show the importance of preprocessing in model training.

Alvi et al [10] built a six class corpus from 5126 comments from different facebook pages. Their method involved using both linguistic feature like n-gram, readability score with quantitative like word count and syllables count. Using these methods their GRU based deep neural network achieved an accuracy of 70.1%. The extracted features also reveal that n-gram tokens and word count were most influential while hashtags didn't have much importance. This gives a rare insight of Bengali comments classification.

Unlike Puja and Alvi Zubair et al. [12] proposed a hybrid framework that combines deep learning with fuzzy logic to evaluate customer feedback. The system firstly uses a BiLSTM with attention mechanism to classify positive and negative tweets. Then

it uses fuzzy logic to understand if customer was satisfied. This approach makes it more accurate in determining feedbacks. Their BiLSTM model achieved an accuracy of 92.86% with strong precision and recall. The fuzzy logic layer makes it more transparent. This shows the usability of these models in different application and usage.

Abdullah et al [12] combined text based feature with users metadata such as location, time, gender to build a robust model. Using WEKA they found SVM was consistent in outperforming other models with the highest accuracy of 97.27%. This adds demographic context to models which boosted the result. This gives another insight into cyberbullying detection on Bengali media.

Wahid et al. [13] also built deep learning model for context capture. They used a corpus of 10000 comments and labelled them into four category. Their RNN model with LSTM units were able to achieve high accuracy. They used word2vec embeddings that helped capture semantic relation while LSTM units capture long term dependencies. It was able to outperform traditional models like SVM and Naïve Bayes. It highlights the importance of embeddings in model perfection.

Drovo et al [14] took a different approach to Named Entity Recognition. They trained model on a 10000 word corpus with 7 entity types. Their approach to model building involved combining a Hidden Markov model with rule based system. They did it using regular expression. The rule based component pre defined entity before passing it to HMM as inputs. They tested two different setup, one where both system ran together and another where rule based tagging followed HMM prediction. Their second method achieved a higher f1 score of 71.59%.This shows how the combination of statistical and rule based method can improve result.

Like Wahid et al [13], Rahman et al [16] used word2vec based neural architecture. They labelled a dataset of 11000 sentence into three classes. After preprocessing skip gram and CBOW was use to generate word embeddings. These embeddings create embedding matrix and calculates sentence level polarity scores. The skip gram model achieved 75% accuracy while CBOW went up to 70%. This approach shows strong semantic clustering and low loss. It highlights the value of hyperparameter tuning and preprocessing.

Khan et al [17] collected a large number of raw data about 63000 comments. From this 10000 were filtered and 3000 was manually labelled with 7 emotion classes. They applied TF-IDF for feature extraction and used four machine learning model and a neural network. Among them SVM performed the best with 62% accuracy on 7 class and 73% on 3 class model. However it struggled with underrepresented class like Religious and Abusive. This study again proves the consistency of SVM in Bangla language classification.

Like other CNN-LSTM hybrid models, Kamyab et al. [18] proposed ACL-SA model. However it used BiLSTM instead of LSTM on hybrid model. Attention mechanism

and gaussian noise and dropout regularization helped reduce overfitting. The input combines TF-IDF weighting and pre trained GloVe embeddings. This makes the model use both statistical and semantic features.

Bhowmik et al. [19] designed a domain specific lexicon dictionary for Bengali texts. They also introduced a rule based scoring algorithm called BTSC. The BTSC assigns sentiment scores based on POS tagging, punctuation and context modifiers. After preprocessing the data TF-IDF was used to extract feature and used in classical machine learning algorithms. Among them SVM with bi gram feature reached the highest 82.21% accuracy.

Prottasha et al [20] used transfer learning with fine tuned Bangla BERT model. It was trained on BanglaLM corpus and combined with a hybrid CNN-BiLSTM model for contextual and phrase capturing. They compared their model with other traditional models like SVM,LDA,ANN and it outperformed them with 94.15% accuracy. It highlights the use of transformer based models in Bangla NLP.

Haque et al [21] collected a dataset of 49178 tweets related to suicide. They tested both ML and DL models on the dataset and found out that they have similar accuracy. However, DL models such as BiLSTM were slightly better and more consistent than machine learning models. CountVectorizer for ML and word embeddings for DL models made the models more accurate. This again proves the consistency and reliability of DL models.

Like Haque et al. , Chowdhury et al [22] also tested multiple ML and DL models on 4000 social media comments. Their preprocessing included tokenization, stop word removal and stemming. CountVextorizer and TF-IDF was used to extract features. In their study they found SVM with TF-IDF achieved an accuracy of 88.90%. 5 fold cross validation was used to validate the models.

Table 2.1: Literature Review Summary.

Author (s)	Year	Title	Methodology	Key Findings
Amit et al. [1]	2021	Bangla hate speech detection on social media using attention-based recurrent neural network	Encoder decoder deep learning (CNN + LSTM/GRU/Attention), TF-IDF, word embedding	Attention based model achieved highest accuracy (77%) in Bangla hate speech classification
Md. Reazul et al. [2]	2021	DeepHateExplainer: An Explainable Approach for	Ensemble of transformer-based models (Bangla BERT, mBERT, XLM-	Achieved 88% F1-score;

		Hate Speech Detection in Bengali	RoBERTa); SA & LRP for interpretability; ML & DNN baselines	
Rezaul et al. [4]	2021	Multi-class sentiment classification on Bengali social media comments using machine learning	ML & DL models; CLSTM with embeddings	CLSTM achieved 85.8% accuracy, F1-score 0.86
Nauros et al. [5]	2021	Hate Speech Detection in the Bengali Language: A Dataset and Its Baseline Evaluation	SVM, LSTM, BiLSTM with Word2Vec, FastText, BengFastText	SVM performed better than neural networks (87.5% accuracy)
Abdul et al. [6]	2018	Detecting Abusive Comments in Discussion Threads Using Naive Bayes	Translated to English then used preprocessing and BOW and trained on Naïve Bayes	Naive Bayes showed strong performance on translated data (Accuracy 80.57%)
Nayan et al [7]	2019	Toxicity Detection on Bengali Social Media Comments using Supervised Models	Used machine learning and deep learning model with TF-IDF and word embeddings	Deep learning models achieve 10% better result than traditional ML. CNN being highest with 95.3%
Puja et al [8]	2019	Threat and Abusive Language Detection on Social Media in Bengali Language	Used MNB, SVM and CNN-LSTM with TF-IDF and embedded vectors	Linear SVM achieved 78% accuracy, deep learning model does better with more data
Estiak et al.[9]	2019	A Deep Learning Approach to Detect Abusive Bengali Text	Compared ML and DL (ANN, RNN-LSTM) using CountVectorizer, TF-IDF and word embedding	RNN-LSTM hybrid achieved highest accuracy of 82.2%

Alvi et al [10]	2019	Hateful Speech Detection in Public Facebook Pages for the Bengali Language	Compared ML and GRU-based DNN using linguistic and quantitative features	GRU model achieved 70.1% accuracy, features show n-gram and word count most important
Zubair et al. [11]	2020	Senti-eSystem: A Sentiment-Based eSystem Using Hybridized Fuzzy and Deep Neural Network for Measuring Customer Satisfaction	BiLSTM with attention mechanism were used. Applied fuzzy logic.	BiLSTM with attention mechanism achieved 92.86% accuracy outperforming other models.
Abdullah et al [12]	2018	Social Media Bullying Detection Using Machine Learning on Bangla Text	Compared NB, SVM, J48 and KNN on Bangla social media posts; included user metadata	SVM achieved the highest accuracy of 97.27%
Wahid et al. [13]	2019	Cricket Sentiment Analysis from Bangla Text Using RNN with LSTM	Combined RNN and LSTM using word2vec embeddings	LSTM outperformed SVM and NB with 95% accuracy
Drovo et al. [14]	2019	Named Entity Recognition in Bengali Text Using Merged HMM and Rule Base Approach	Combined Hidden Markov Model (HMM) with Regular Expression-based rule system	Rule based detection improved accuracy on patterned entities
Rahman et al [16]	2022	A Dynamic Strategy for Classifying Sentiment From Bengali Text by Utilizing Word2vector Model	Used word2vec on three classes. Polarity scoring and embedding matrix tuning was done	Skip gram and CBOW achived similar performance. With skip gram being 75% accurate. IT outperformed CNN,LSTM and GRU
Khan et al [17]	2021	Sentiment Analysis on Bengali Facebook Comments To Predict Fan's Emotions	Applied TF-IDF and trained SVM,RF,KNN, NB,NN classifier	SVM achieved highest accuracy of 62% on 7 class emotion.

		Towards a Celebrity		
Kamyab et al. [18]	2021	Attention-Based CNN and Bi-LSTM Model Based on TF-IDF and GloVe Word Embedding for Sentiment Analysis	Combined CNN , Bi-LSTM and attention to propose ACL-SA model. TF-IDF and Glove embeddings with gaussian dropout was used	The hybrid model achieved an accuracy of 94.5% on SD4A dataset.
Bhowmik et al. [19]	2021	Bangla Text Sentiment Analysis Using Supervised Machine Learning with Extended Lexicon Dictionary	Developed domain specific weighted lexicon dictionary. Proposed BTSC rule based scoring algorithm and applied TF-IDF,n-grams on SVM	SVM using Bi-grma and TF-IDF achieved the highest accuracy of 82.21%
Prottasha et al. [20]	2022	Transfer Learning for Sentiment Analysis Using BERT Based Supervised Fine-Tuning	Fine-tuned Bangla-BERT and integrated with CNN-BiLSTM.	Achieved 94.15% accuracy with Bangla-BERT + CNN-BiLSTM
Haque et al. [21]	2022	A Comparative Analysis on Suicidal Ideation Detection Using NLP, Machine and Deep Learning	Using a collection of 49,178 tweets both ML and DL models were tested	BiLSTM achieved highest accuracy of 93.6% with RF being similar. DL model were consistent with accuracy.
Chowdhury et al.[22]	2019	Analyzing Sentiment of Movie Reviews in Bangla by Applying Machine Learning Techniques	Tokenized,stemmed raw data before using on both machine learning and deep learning models	SVM with TF-IDF reached the highest accuracy of 88.90%

2.3 Gap Analysis

Table 2.2: Gap Analysis

Name	Language	DataSet	Classes	result	Implementation
Amit et al. [1]	Bangla	7,425	7 (Hate Speech, Aggressive, Religious Hatred, Ethnical, Religious, Political, Suicidal)	Accuracy 77% (Attention-RNN)	None
Md. Reazul et al. [2]	Bangla	5000	4 (Political, Religious, Geopolitical, Personal)	F1 score 0.87	None
Rezaul et al. [4]	Bangla	42036	4 (Sexual, Religious, Political, Acceptable)	Accuracy 85.8% (CLSTM)	Web
Nauros et al. [5]	Bangla	30000	2 (Hate,non-hate)	Accuracy 87.5%, F1 score 0.91	None
Abdul et al. [6]	English	2665	2 (Abusive, non-abusive)	Accuracy 80.57%, F1 score 0.39	None
Nayan et al [7]	Bangla	10219	2 (toxic, nontoxic)	Accuracy 95.3%	None
Puja et al [8]	Bangla	5644	2 (Abusive, Non-Abusive)	Accuracy 78% (SVM)	None
Estiak et al.[9]	Bangla	4700	7 (Slang, Religious Hatred, Personal Attack, Political, Antifeminism, Positive, Neutral)	Accuracy 82.2% (RNN+LSTM)	None
Alvi et al [10]	Bangla	5126	6 (Hate Speech, Communal Attack, Inciteful, Religious Hatred, Political, Religious)	Accuracy 70.1% (GRU)	None

Zubair et al. [11]	English	11541	2 (Positive, Negative)	Accuracy 92.86%, F1 score 0.93 (BiLSTM +attention+fuzzy logic)	None
Abdullah et al [12]	Bangla	2400	2 (Bullied, Not Bullied)	Accuracy 97% (SVM)	None
Wahid et al. [13]	Bangla	10,000	3 (Positive, Negative, Neutral)	Accuracy 95% (LSTM)	None
Drovo et al. [14]	Bangla	10000	7	F1 score 71.59%	None
Rahman et al [16]	Bangla	11,000	3 (Happy, Angry, Excited)	Accuracy 75%	None
Khan et al [17]	Bangla	10,000	7 (Happy, Sad, Angry, Surprised, Excited, Abusive, Religious)	Accuracy 62% (SVM)	None
Kamyab et al. [18]	English	1,120,000	2–3 (Positive, Negative, Neutral)	Accuracy 94.53%	None
Bhowmik et al. [19]	Bangla	5,137	3 (Positive, Negative, Neutral)	Accuracy 82.21% (SVM)	None
Prottasha et al. [20]	Bangla	8952	2 (Positive, Negative)	Accuracy 94.15% (BERT + CNN-BiLSTM)	None
Haque et al. [21]	English	49,178	2 (Suicidal, Non-Suicidal)	Accuracy 93.6% (BiLSTM)	None
Chowdhury et al.[22]	Bangla	4,000	2 (Positive, Negative)	Accuracy 88.90% (SVM), 82.42% (LSTM)	None

2.4 Summary

Reviewing multiple studies reveal that some of the models were consistent regardless of datasets. In classical machine learning models SVM was consistent throughout multiple studies. However deep learning models were the best among all, specially the CNN-BiLSTM hybrid model. The deep learning models as an individual outperforms machine learning models as well. The strength of CNN-

BiLSTM hybrid come from contextual and phrase capturing. Aside from these models transformer based models like Bangla BERT also performed exceptionally well. Many studies also contributed dataset, lexicon and website for usage. This goes to show that there are more to study in the field of growing Bengali NLP.

Chapter 3

Research Methodology

3.1 Requirement Analysis & Design Specification

After preprocessing Bengali comments using natural language processing methods, we created a number of machine learning, deep learning models to examine topics, sentiments.

3.1.1 Overview

Our methodology follows most of the common conventions in its initial stages. At first, we collected our required data from various social media platforms. Each data was manually labelled with the most suitable topic and sentiment for it.

We used an open-source python package named 'bnltk' created by Asraf Patoary from his GitHub repository. It provides various functionalities to make Bengali data more suitable for training. Any present imbalance in the data was handled using oversampling before encoding them and training the models.

From testing different models we got to a threshold of performance that was hard to overcome no matter which model we used. That is why we proposed a better method compared to training single models that can overcome that threshold to a certain extent and open up new possibilities for even better outcomes in the future.

3.1.2 Proposed Methodology

The Proposed architecture combines the strength of multiple deep learning model to one. This starts with collecting raw data from various social media (facebook, youtube,twiiter) using octoparse 8 and google api. The collected data were processed to contain only Bengali comments and labelled accordingly.

The dataset were preprocessed and split into training and testing data. Class imbalance were fixed through oversampling. The proposed model was made by combining the prediction of CNN and Bilstm and stacking SVM on top of it as the final judge. This has proven to be more effective than regular models.

The model was evaluated thoroughly. Evaluation metrics including accuracy, precison, recall, f1-score were observed to evaluate along with ROC and PR curve for the final output. Each individual model was also tested to see their performance. A website were also created with flask and JavaScript in order to implement the model.

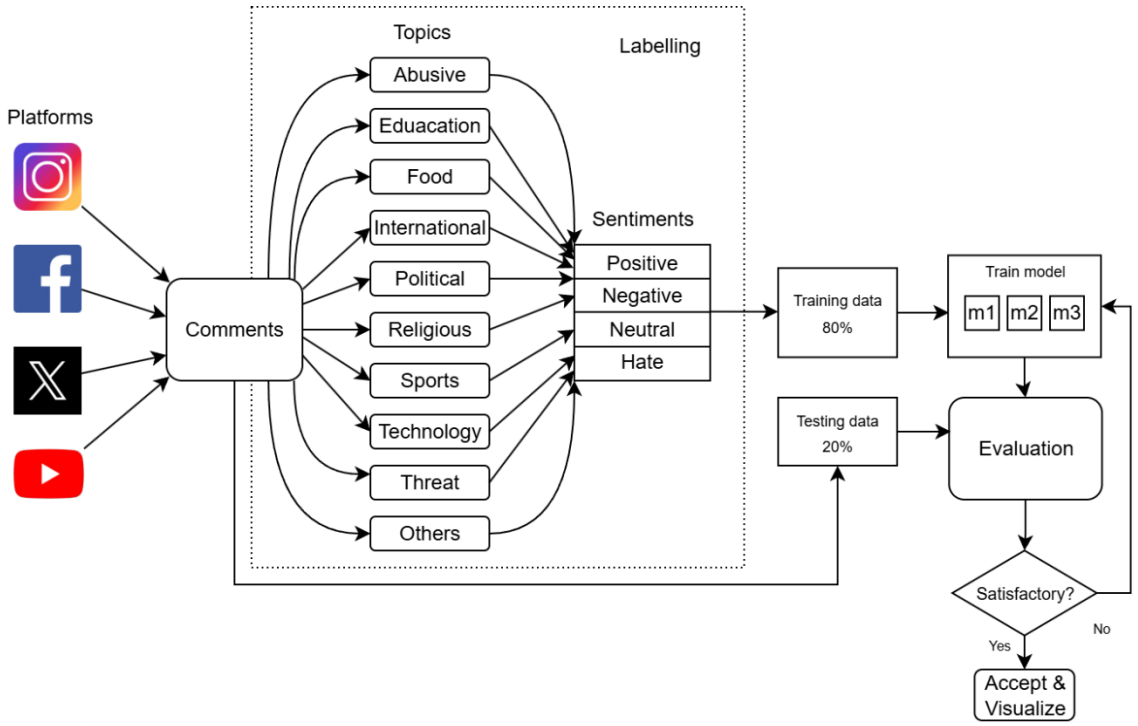

Figure 3.1: Methodology overview

As displayed in Figure 3.1, first the data had been collected from four different social media platforms. Then they were each labelled with one of the 10 Topic classes and one of the four sentiment classes manually before moving on to the next step. The data was then split into two portions, 80% for training and the remaining 20% for testing. The models were then trained on the training data and evaluated based on the testing data. If the result for a certain model was not satisfactory, then that was skipped and another model was trained. The models that achieved satisfactory results were accepted and their performance was visualized using relevant charts and graphs.

3.1.3 UI Design

Figure 3.2 is the initial view of our web app. It presents two options to the users. The first input field is to analyze a single comment and the second one is for analyzing a YouTube video.

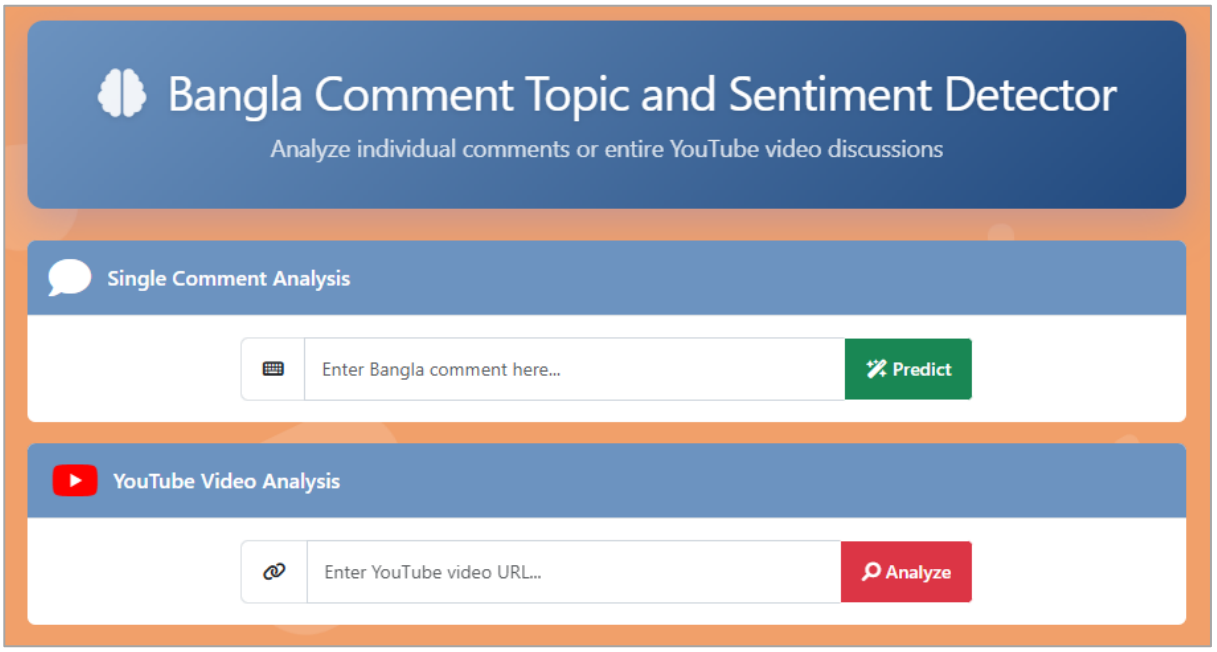

Figure 3.2: Website Homepage

Single Comment Analysis:

To use the Single comment analyzer, the user has to manually input a single Bengali comment and press the Predict button. This will send the comment to one of our trained models and the model will detect topic and sentiment of the comment and show it as output. An example of it's usage is displayed in Figure 3.3.

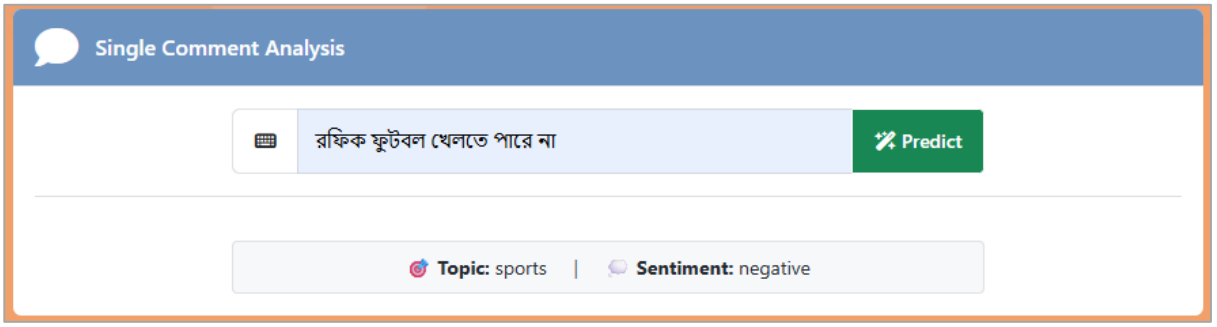

Figure 3.3: Single comment analysis

YouTube Video Analysis:

The YouTube video analyzer option has a bit more complex working method in the backend but still very user-friendly. The only complex thing the user has to do is creating an API key from Google cloud services for using YouTube Data API v3. After that, all the user has to do is input the URL of the YouTube video they want to analyze and click the analyze button. This will fetch the comments of the video and then analyze each of them

and show the results with proper visualization and details to the user as shown in Figure 3.4.

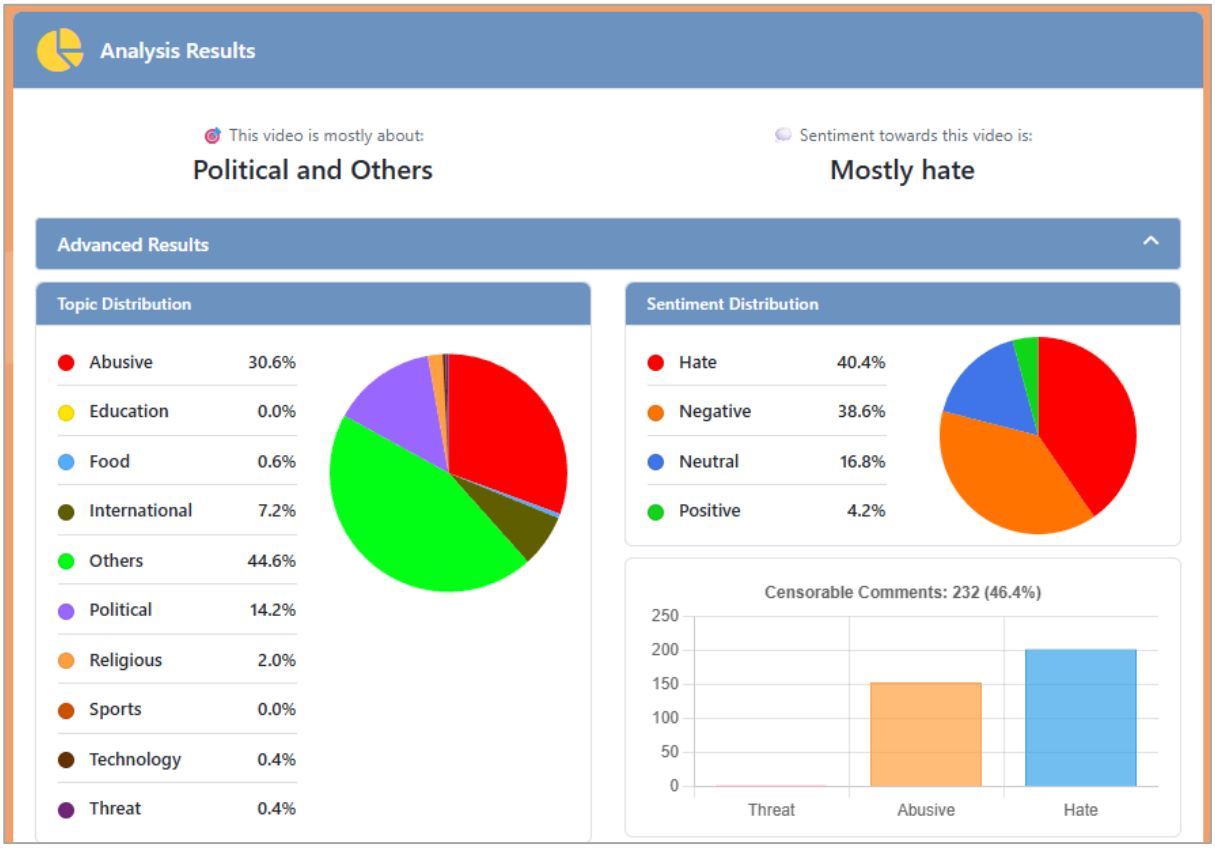

Figure 3.4: YouTube video analysis

3.2 Detailed Methodology and Design

3.2.1 Data Collection

In order to collect a dataset on Bengali social media comments for topic and toxicity classification we tried various approaches. The first approach was using Octoparse 8, a web scrapping tool. This tool worked as intended but the free plan had significant limitations. This restricted our collection progress.

To overcome the difficulty, we manually gathered some comments from Facebook, Instagram, YouTube. Another better method of data collection were found in our research. We used YouTube Data API v3 through google AI studio for increase efficiency in our task. All of the data were taken from public posts and video and in accordance with terms of service of the media platform.

The Focus were only in comments written in Bangla and containing fewer than 250 characters. We filtered out comments in other language, emoji or inexplainable comments through API call in google colab. In order to improve quality we removed duplicate data.

The Dataset of 7,725 lines were manually labelled by topic and sentiment behind the comment. This dataset will be a valuable asset in the field of Bengali natural language processing.

3.2.2 Dataset Description

The Dataset consists on 7,725 Bangla comments from various platform. Each comment is labelled by topic (10 classes) and by Sentiment (4 classes). The dataset were split into 80% training and 20% testing data. From the training data it was further split to 20% validation data. The data compromises of 10 Topic classes (others, abusive, political, religious, international, education, food, sports, technology, threat). It also has 4 Sentiment Class (positive, negative, neutral, hate).

Table 3.1: Preview of dataset

	Date	Comments	Topic	Sentiment
0	12/27/2024	আমার এলাকায় বনশ্রীতে খবর বেশি একটা ভালো না	others	negative
1	12/24/2024	৬৫ পদের মশলা দেখছে ও একসাথে! বিশ্ব চাপাবাজ! আমার বাসার যে রেয়ার মশলা আইটেম আছে তা হিসাব করলেও সর্বোচ্চ ২৫-৩০ আইটেমের মশলা হতে পারে।	food	negative
2	12/11/2024	ফাহিম ভাইয়ের ব্লগ আমার ঘরনি পছন্দ করে	others	positive
3	12/17/2024	বাচ্চাদের সাথে আপনার ব্যবহার খুব সুন্দর লাগলো	others	positive
4	12/12/2024	মাংসে সর্বোচ্চ ৩০ প্রকারের মসলা যেটা ব্যবহার করা যায়। আর উনি বাদাম কে মসলা বলে চালায় দিল । সব ধরনের মসলা গরুর গোস্তে ব্যবহার করা যায় না।	food	negative

The dataset contains 2731 others comment and 1098 political comments. The sentiment count for negative has the highest with 3463 and for positive 1501.

Table 3.2: Topic distribution in dataset

Topic	Count
Abusive	801
Education	458
Food	439
International	847
Political	1098
Religious	663
Sports	355
Technology	239

Threat	94
Others	2731

Table 3.3: Sentiment distribution in dataset

Sentiment	Count
Positive	1501
Negative	3463
Neutral	1907
Hate	854

Statical Analysis of the data:

- The dataset consists of total 7725 rows and 4 columns
- The dataset contains a total of 6180 samples for training
- The dataset holds a total of 1545 samples for testing
- The labels are divided into 10 classes for topic and 4 classes for sentiment

The Topic distribution has 35.4% of others class making it the highest. The class with least number of samples come from threat class.

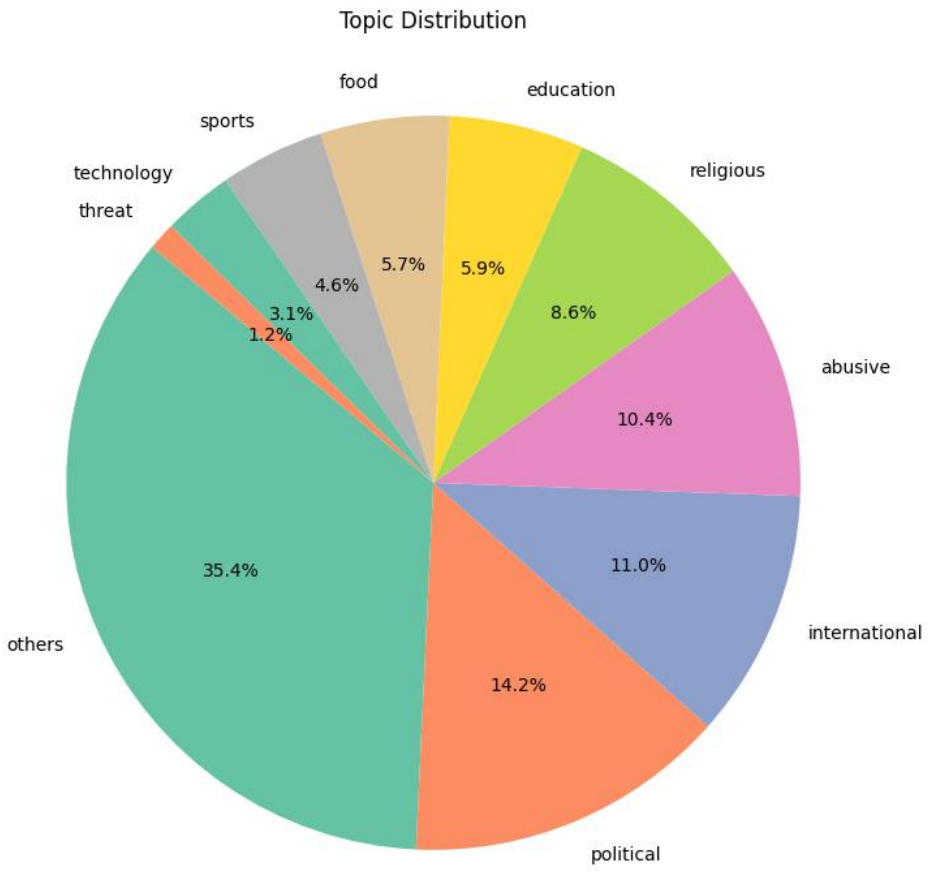

Figure 3.5: Topic Distribution

The Sentiment distribution sees mostly negative comments. Hate class has the lowest

amount of sample but sufficient enough compared to other classes.

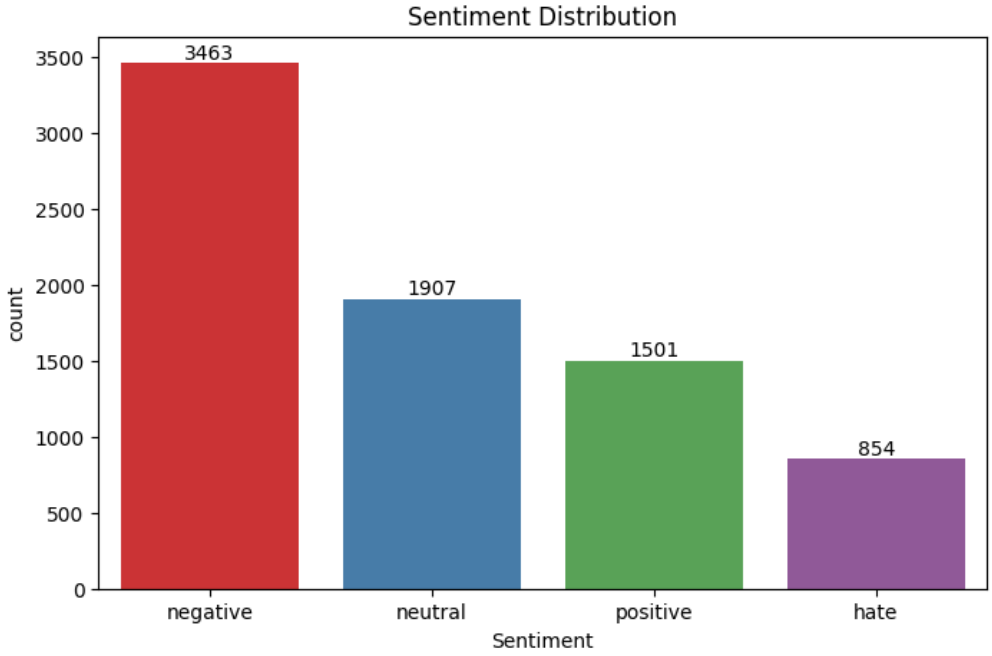

Figure 1.6: Sentiment Distribution

The Combined dataset shows 1190 samples of negative comments in others category. Most of the hate comments contain profanities classified in abusive topic.

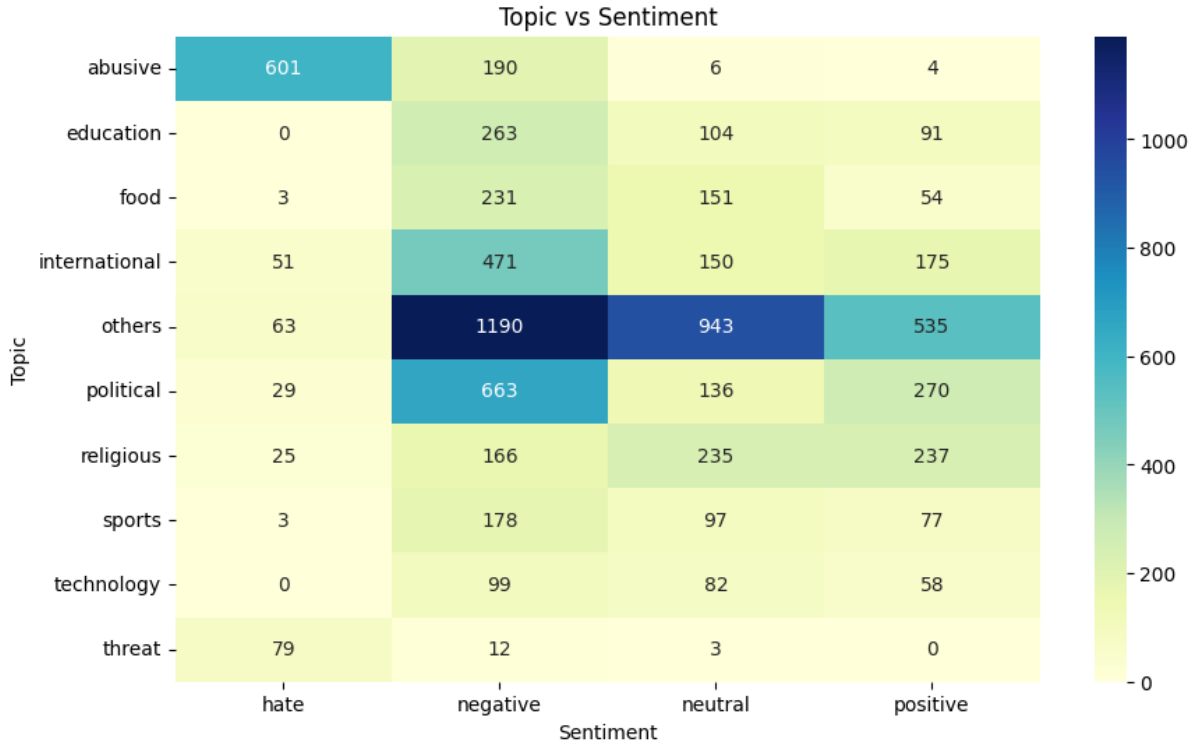

Figure 3.7: Topic vs Sentiment confusion matrix

3.2.3 Data Preprocessing

While collecting the data we checked for null and duplicate values. Commas and quotes

were also removed to reduce noise. Emoji and English texts were also filtered out. Handled the new line by replacing them with spaces. The data collection collected were bound to 250 character max to avoid spam comments.

The preprocessing were done using blntk python library. Using the library invalid characters were removed. Then it normalizes white spaces.

The normalized text is tokenized and stemmed according to the bnltk library. Normalization is the process where the text is simplified and standardized.

Stemming is a process where the tokenized words are replaced by their root form. This helps the model identify words similar words. The separated words are then added back.

Table 3.4: Data preprocessing steps

Original Comment	After Cleaning	Tokenized	Stemmed	Final text
এই বিশ্ববিদ্যালয়ের শিক্ষার মান এবং পরিবেশ সত্যিই প্রশংসনীয়, যা শিক্ষার্থীদের উন্নয়নে সহায়ক।	এই বিশ্ববিদ্যালয়ের শিক্ষার মান এবং পরিবেশ সত্যিই প্রশংসনীয় যা শিক্ষার্থীদের উন্নয়নে সহায়ক	[“এই”, “বিশ্ববিদ্যালয়ের”, “শিক্ষার”, “মান”, “এবং”, “পরিবেশ”, “সত্যিই”, “প্রশংসনীয়”, “যা”, “শিক্ষার্থীদের”, “উন্নয়নে”, “সহায়ক”]	[“এই”, “বিশ্ববিদ্যালয়”, “শিক্ষা”, “মান”, “এবং”, “পরিবেশ”, “সত্যি”, “প্রশংসনীয়”, “যা”, “শিক্ষার্থী”, “উন্নয়ন”, “সহায়”]	এই বিশ্ববিদ্যালয় শিক্ষা মান এবং পরিবেশ সত্যি প্রশংসনীয় যা শিক্ষার্থী উন্নয়ন সহায়
প্রযুক্তির অগ্রগতির ফলে আমাদের দৈনন্দিন জীবনযাত্রা অনেক সহজ এবং গতিশীল হয়ে উঠেছে।	প্রযুক্তির অগ্রগতির ফলে আমাদের দৈনন্দিন জীবনযাত্রা অনেক সহজ এবং গতিশীল হয়ে উঠেছে	[“প্রযুক্তির”, “অগ্রগতির”, “ফলে”, “আমাদের”, “দৈনন্দিন”, “জীবনযাত্রা”, “অনেক”, “সহজ”, “এবং”, “গতিশীল”, “হয়ে”, “উঠেছে”]	[“প্রযুক্তি”, “অগ্রগতি”, “ফলে”, “আমাদের”, “দৈনন্দিন”, “জীবন”, “অনেক”, “সহজ”, “এবং”, “গতিশীল”, “হয়”, “উঠ”]	প্রযুক্তি অগ্রগতি ফলে আমাদের দৈনন্দিন জীবন অনেক সহজ এবং গতিশীল হয় উঠ
এই সেবার মান ১০০% খারাপ! আমি একেবারে হতাশ।	এই সেবার মান খারাপ আমি একেবারে হতাশ	[“এই”, “সেবার”, “মান”, “খারাপ”, “আমি”, “একেবারে”, “হতাশ”]	[“এই”, “সেবা”, “মান”, “খারাপ”, “আমি”, “একেবারে”, “হতাশ”]	এই সেবা মান খারাপ আমি একেবারে হতাশ
শিক্ষার্থীদের জন্য একটি ভালো পাঠ্যক্রম এবং দক্ষ শিক্ষক	শিক্ষার্থীদের জন্য একটি ভালো পাঠ্যক্রম এবং দক্ষ শিক্ষক	[“শিক্ষার্থীদের”, “জন্য”, “একটি”, “ভালো”, “পাঠ্যক্রম”, “এবং”, “দক্ষ”,	[“শিক্ষার্থী”, “জন্য”, “একটি”, “ভালো”, “পাঠ্য”, “এবং”, “দক্ষ”,	শিক্ষার্থী জন্য একটি ভালো পাঠ্য এবং দক্ষ

অত্যন্ত গুরুত্বপূর্ণ, বিশেষ করে প্রাথমিক পর্যায়ে।	অত্যন্ত গুরুত্বপূর্ণ বিশেষ করে প্রাথমিক পর্যায়ে	“শিক্ষক”, “অত্যন্ত”, “গুরুত্বপূর্ণ”, “বিশেষ”, “করে”, “প্রাথমিক”, “পর্যায়ে”]	“শিক্ষক”, “অত্যন্ত”, “গুরুত্বপূর্ণ”, “বিশেষ”, “কর”, “প্রাথমিক”, “পর্যায়”]	শিক্ষক অত্যন্ত গুরুত্বপূর্ণ বিশেষ কর প্রাথমিক পর্যায়
আন্তর্জাতিক সম্পর্ক উন্নয়নের জন্য কূটনৈতিক দক্ষতা এবং সাংস্কৃতিক বোঝাপড়া অপরিহার্য।	আন্তর্জাতিক সম্পর্ক উন্নয়নের জন্য কূটনৈতিক দক্ষতা এবং সাংস্কৃতিক বোঝাপড়া অপরিহার্য	[“আন্তর্জাতিক”, “সম্পর্ক”, “উন্নয়নের”, “জন্য”, “কূটনৈতিক”, “দক্ষতা”, “এবং”, “সাংস্কৃতিক”, “বোঝাপড়া”, “অপরিহার্য”]	[“আন্তর্জাতিক”, “সম্পর্ক”, “উন্নয়ন”, “জন্য”, “কূটনীতি”, “দক্ষতা”, “এবং”, “সংস্কৃতি”, “বোঝা”, “অপরিহার্য”]	আন্তর্জাতিক সম্পর্ক উন্নয়ন জন্য কূটনীতি দক্ষতা এবং সংস্কৃতি বোঝা অপরিহার্য

3.2.4 Tested models

SVM:

Support Vector machine is a supervised learning model. The model tries to draw a boundary between classes known as a hyperplane. The points close to hyperplane are most focused, these are support vectors. SVM tries to find the biggest gap possible between support vectors. It assigns new data to class based on this hyperplane as shown in figure 3.8.

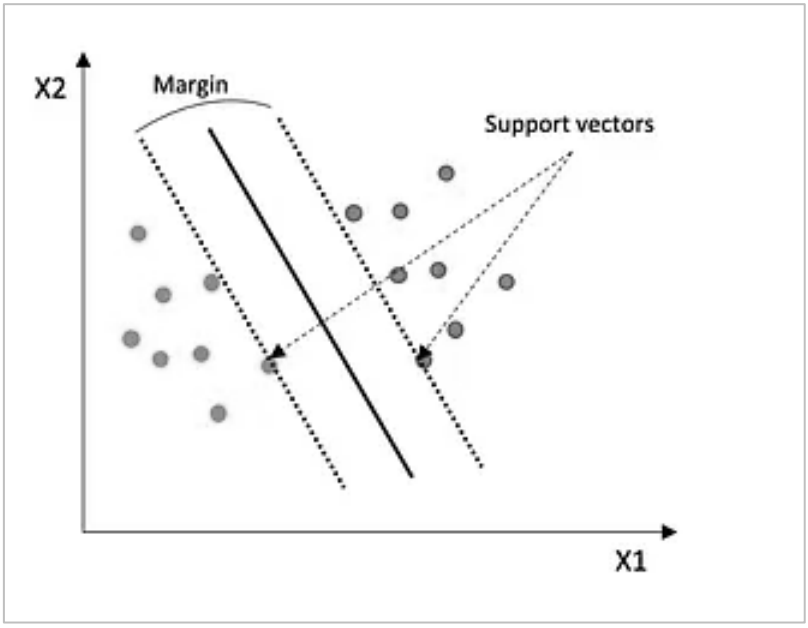

Figure 3.8: SVM architecture

CNN:

Convolutional Neural Network (CNN) is a deep learning model known for its feature extraction (figure 3.9). The model works by converting texts to vectors using embeddings. This embedding creates a matrix that can be multiplied with a convolution filter to get

feature map. The tested model uses bi-gram, tri-gram, four-gram filter for phrase capturing. Each convolution has 128 filters. This helps capture local features. GlobalAveragePooling1D was done to reduce the dimensionality after each convolution. The extracted features are concatenated and run through a dense layer network to make final classification.

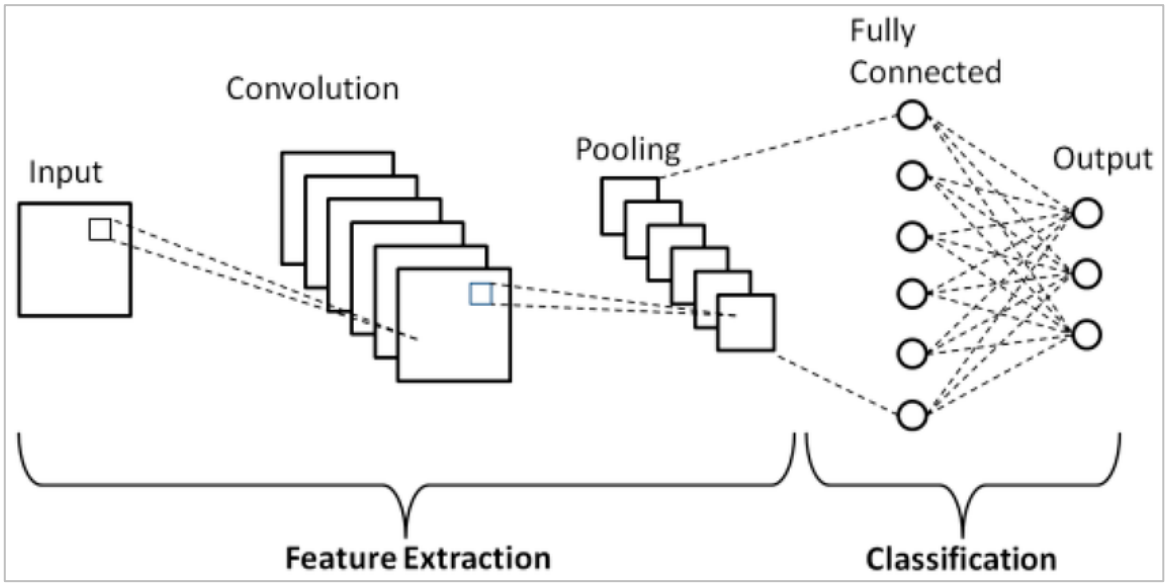

Figure 3.9: Convolutional Neural Network architecture

LSTM:

Long Short-Term Memory (figure 3.10) is a type of RNN for capturing long term dependencies. It uses 3 gates (input, forget, output) to produce results on each node. The input gate controls how much data is added. The forget gate decides what to remember and what to discard. It combines with existing data and updates the memory. This is then passed to next node. Like RNN it shares its weight across same node.

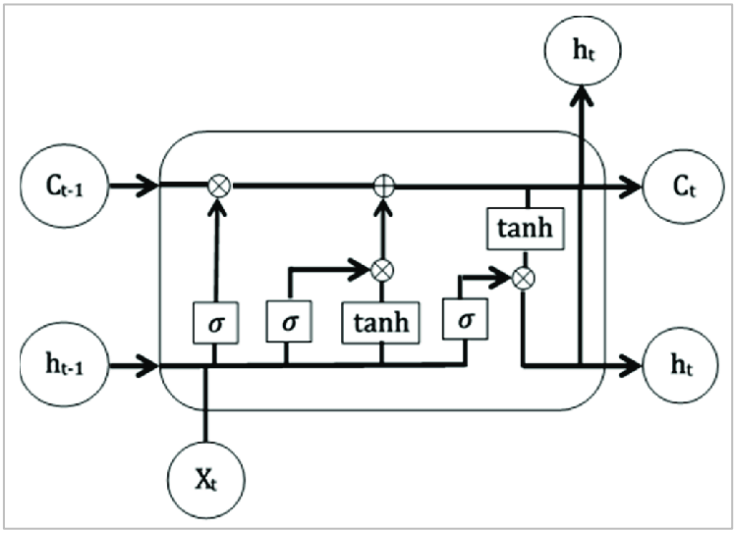

Figure 3.10: LSTM architecture

BiLSTM:

Bi- dirtectional LSTM is an extension of LSTM. It works similar to LSTM. But instead of one direction it uses both forward and backward layer to understand context from both side, visualized with figure 3.11. It combines both hidden states to produce an output. Its good for capturing long term dependency and capturing correct meanings. Its effective in text classification where context is important.

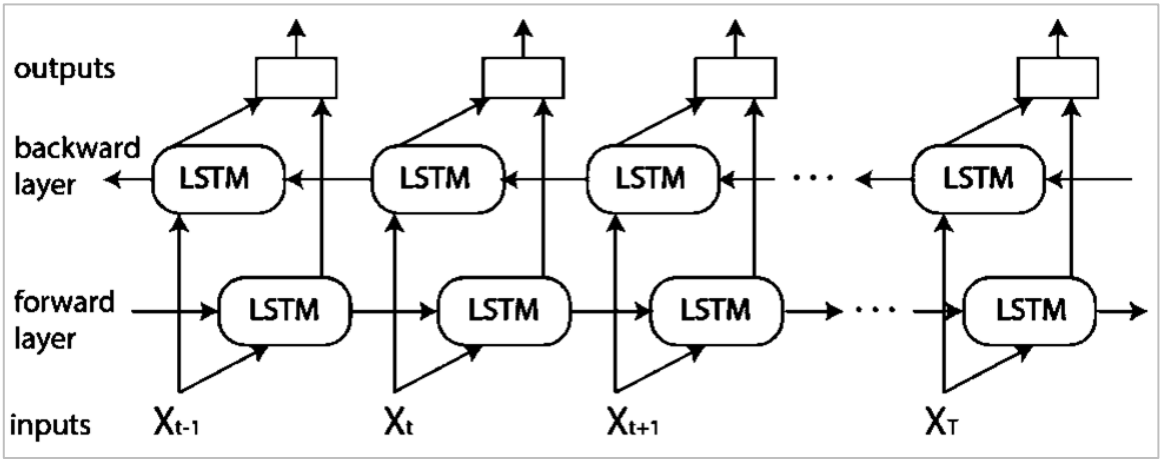

Figure 3.11: BiLSTM architecture

Bangla-BERT:

BanglaBert is a pre trained transformer-based model made for Bengali language. IT uses ELECTRA framework where generator replace input tokens. These tokens are then detected by discriminator (figure 3.12) . This model was trained on a 27.5GB of Bengali text data from Banlga2B+. The tokenizer used in this model is specially optimized for Bengali language. It makes the model more robust to native language.

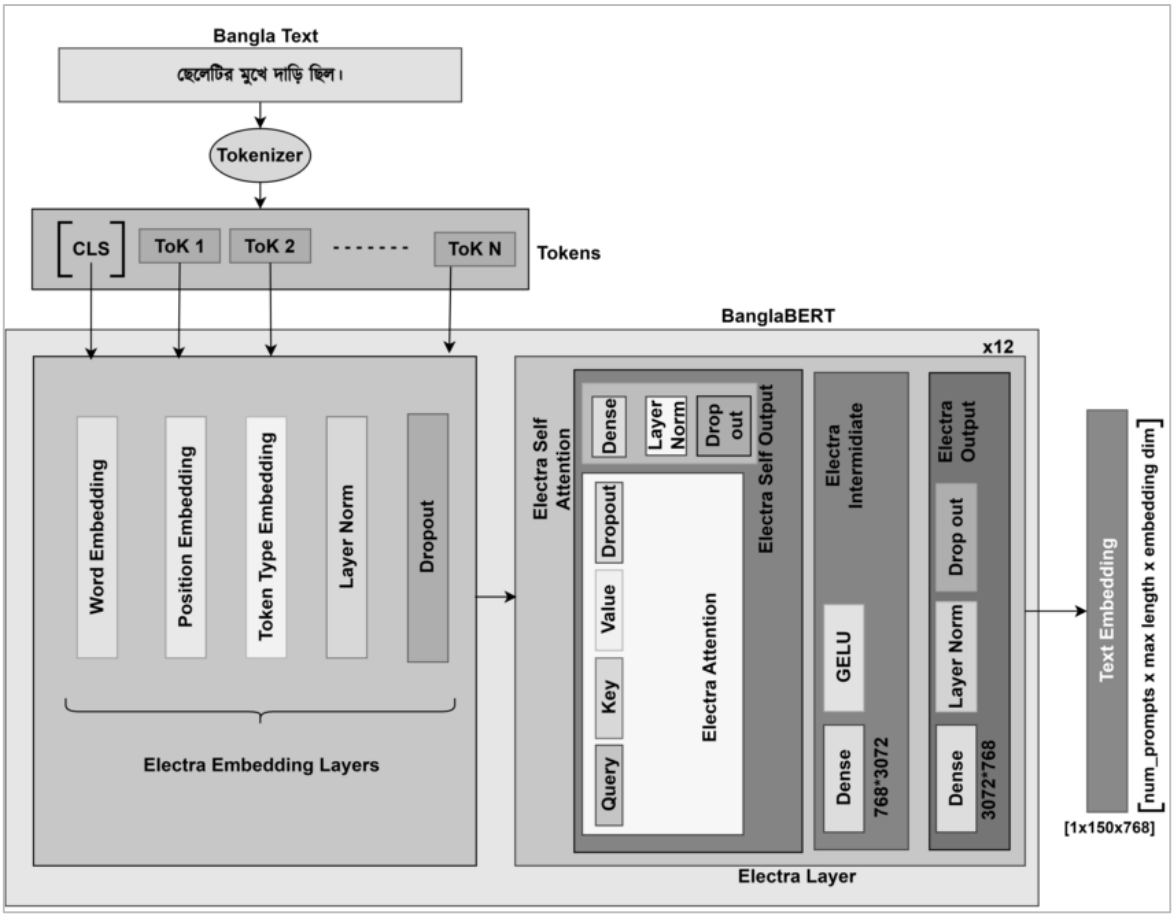

Figure 3.12: Bangla-BERT internal architecture

Multinomial Naive Bayes:

Multinomial Naive Bayes (figure 3.13) is a machine learning model on probabilistic classification. It's based on bayes theorem. It counts how many times a word appears on document. Then each document is converted to word frequency vector. For each class it learns how often a word appears and each class probabilities. For a new document it applies bayes theorem and gets the probabilities of all class. Depending on probabilities it assigns class to it. It can classify unseen data more accurately based on its learning.

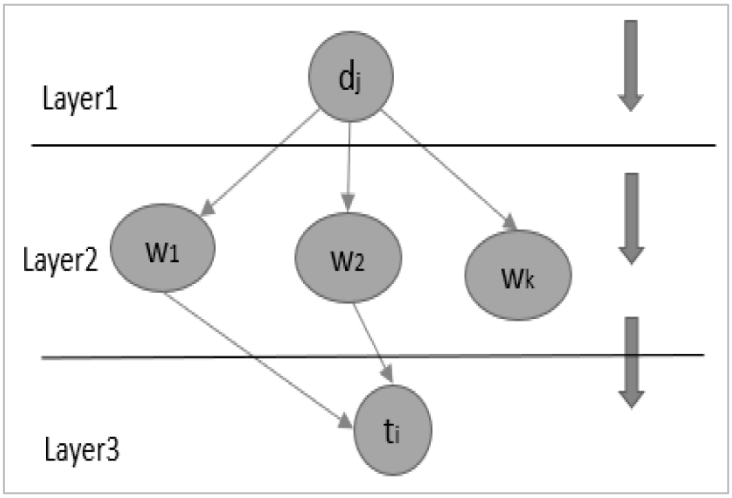

Figure 3.13: Multinomial Naïve Bayes

Logistic Regression:

Logistic regression is a supervised machine learning model used for classification. IT works by converting given data to vectors. In the tested model TF-IDF scores were used. Then it computes the weighted sum of inputs. This input is then passed through an activation function and from this an output is produced. The model calculates error and based loss function it updates the weights accordingly illustrated with figure 3.14.

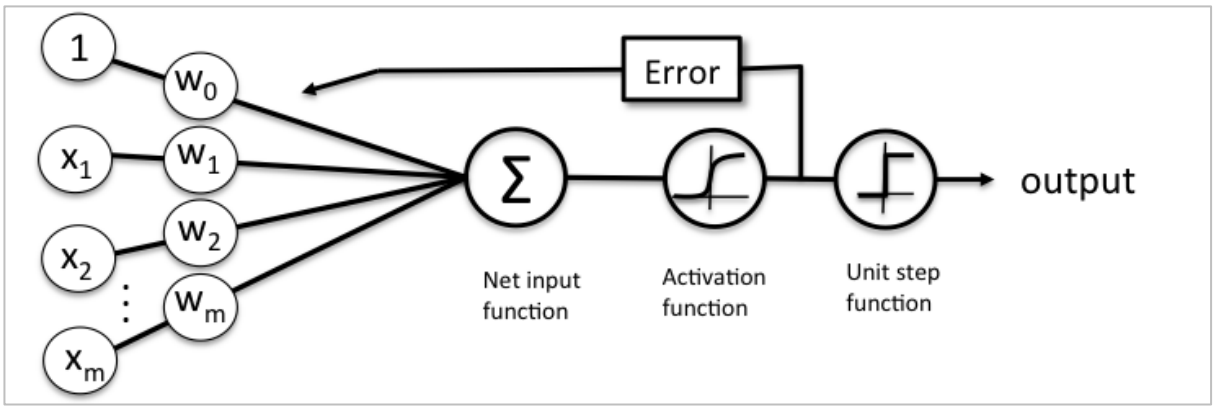

Figure 3.14: Logistic Regression

Random Forest Classifier:

Random Forest Classifier combines multiple decision trees for prediction. Each decision tree is trained on a portion of the training data. Each tree considers a number of features at each split and learns to classify according to these features. For a new input each tree makes a prediction. The input is then classified by majority vote or averaging probabilities from multiple decision tree (figure 3.15).

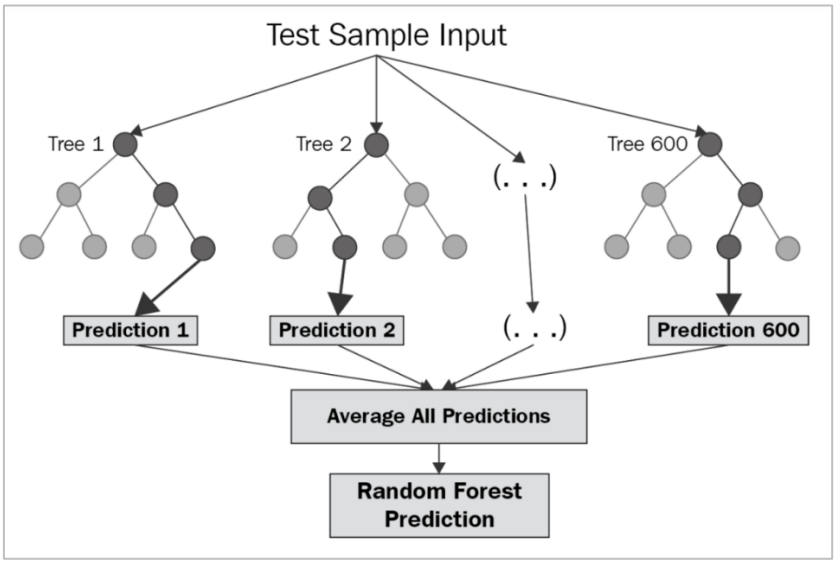

Figure 3.15: Random Forest Classifier

XGBoost:

XGBoost is a machine learning model that does classification using multiple decision trees. It works similarly to random forest but instead of making independent trees it makes them one after another. It uses gradient boosting, fixing errors each step, shown in figure 3.16. The new trees are built after fixing errors on previous one. It has regularization to minimize overfitting. It works well on structured dataset.

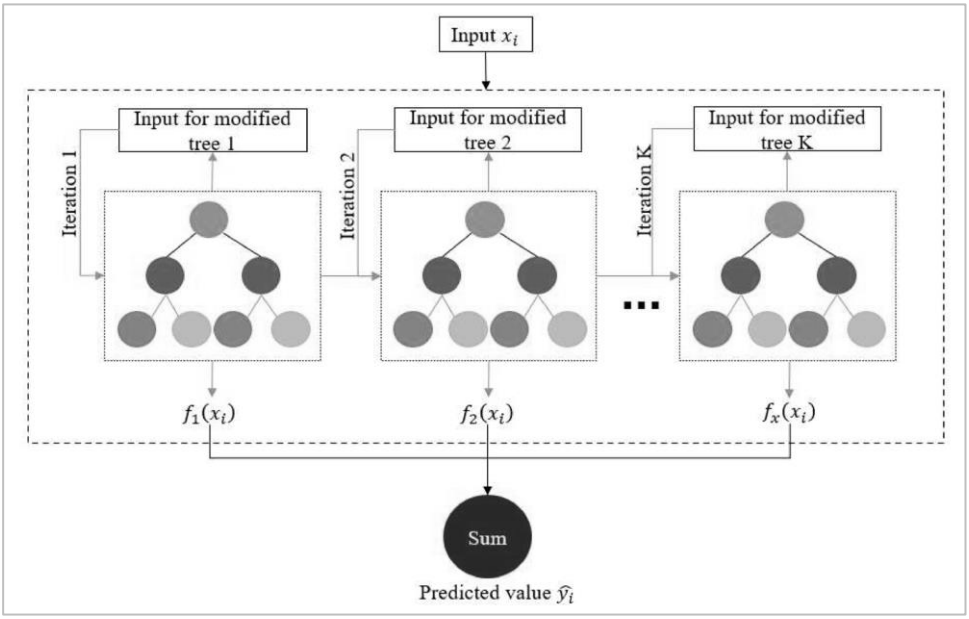

Figure 3.16: XGBoost schematic diagram

3.2.5 Proposed model

The proposed model (figure 3.17) is an ensemble stack model with SVM classifier as meta model. It uses CNN and BiLstm probabilities to train SVM.

The model starts by cleaning out full stop, exclamation mark, numbers from raw data collected. Cleaned comments are then tokenized using bnltk Bengali tokenizer. These comments are split into individual words while tokenizing. The split tokens are reduced to root from through stemming. It helps model recognize similar words and not treat them as a new one. The stemmed tokens are then concatenated together to form a string in order to obtain compatibility with keras.

Some of the classes in dataset have less data than others. These are minority classes that are present in the data. In order to overcome data imbalance RandomOversampler from imbalanced-learn was used. This makes the underrepresented classes like threat, sports, technology be represented while training the model.

This model applies a two stage encoding. The label encoder is used to convert topic and sentiment to integer class. These classes are unique for each label. This is useful for classification reports. These integer classes are then converted to single vector containing only 1's and 0's using onehotencoder. Its needed for the loss function Categorical cross entropy used in the model. This is needed to compare with true one hot label and update weights accordingly.

The dataset is split into 80% training and 20% testing. Since the classes in dataset were imbalanced, stratified split was done to preserve minority class. This guarantees that the after split dataset maintains original class distribution. This helps model train on classes that would otherwise might be accidentally ignored.

The proposed model uses two neural network model and CNN is one of them . CNN is a deep neural architecture that focus on capturing spatial information. Its good for catching n-gram features like phrases. The model captures bigram, trigram and four gram patterns effectively using varying kernel sizes. Each convolution output is converted to single vector using GlobalAveragePooling1D.The outputs are then concatenated and a output probability vector is produced using softmax.

BiLstm is another deep learning model good on capturing sequential data . Bidirectional lstm reads from both forward and backward improving context capturing. Its used with attention layer to capture the informative parts of sequence. The output of this model is converted to probabilities using softmax.

Both probabilities vector from CNN and BiLstm is merged to a single vector. SVM then learns to map from this combined prediction vector to correct class. In this way SVM learns which model to trust more for each class or scenario. The model gives a decision based off of the prediction mapping.

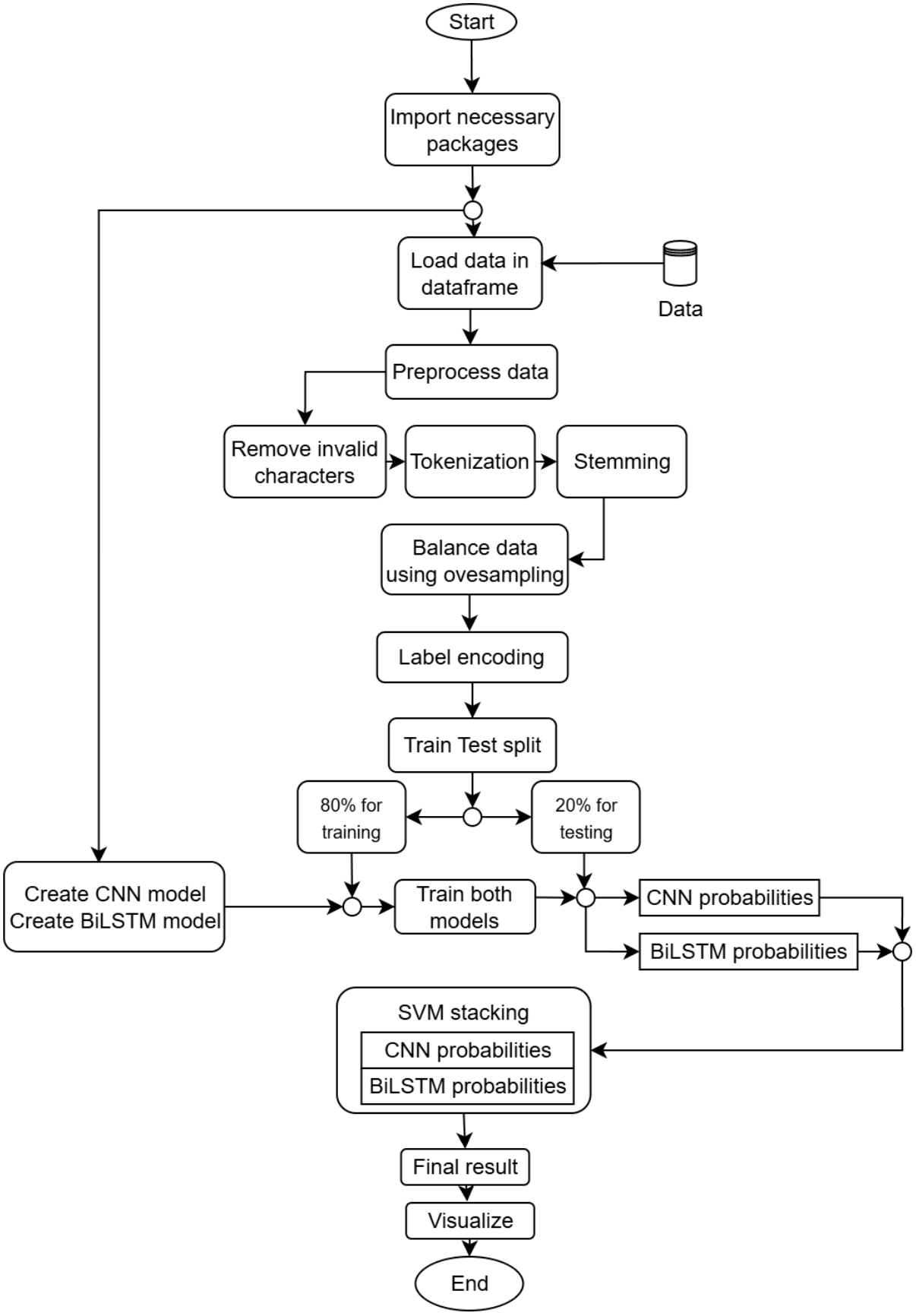

Figure 3.17: Proposed model

The end model is an ensemble model that uses strength of both CNN and BiLstm and gives decision based on it

3.3 Project Plan

The project was implemented in a number of structured phases for systematic completion and evaluation. First, a suitable dataset of Bengali comments was collected, pre-processed with standard NLP techniques such as tokenization, stemming and noise removal to allow for analysis. The modeling phase scoped the implementation of both machine learning algorithms and deep learning architectures in Google Colab by employing different machine learning and deep learning approaches, trying to take advantage of both. Each model was trained, validated and tested using stratified splits to ensure class balance. Hence the performances were compared for accuracy, confusion matrices and ROC/PR curves. The final stage of the project was to analyze the results to assess the best method of classifying topics and sentiment and looking forward as to how the system can be extended towards real world applications of toxicity detection.

3.4 Task Allocation

This table (table 3.5) depicts the timeline of the principal activities in each period of the project,
from week 12 to week 48.

Table 3.5: Task allocation

Tasks	Weeks																		
	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48
Data collection phase																			
Data labelling																			
Model training																			
Create a demo website.																			

3.5 Summary

This research used an experimental process that involved preprocessing a considerable dataset of Bengali text by tokenization, stemming and removing irrelevant characters. Several models were implemented, such as machine learning classifiers, deep learning models. The models were trained, using stratified train-test splits, to ensure each model received similar performance across classes. The results were evaluated with common evaluation metrics of accuracy, confusion matrices, ROC curves and precision-recall curves to help compare results. The use of this kind of methodology provided a solid and fair evaluation of approaches to classify in terms of topic, sentiment and possibly toxicity.

Chapter 4

Implementation and Results

4.1 Environment Setup

4.1.1 Hardware and Runtime

We used Python in Google Collaboratory to run and test all the models in our experiment. It is a cloud based Jupyter Notebook offered by Google. For the deep learning models such as CNN, LSTM, BiLSTM etc. we used a T4 GPU based runtime to make the training faster as they require more computational resources.

4.1.2 Dependencies

Although each model requires different dependencies to run, for almost all the models the same packages were used for data preprocessing. Since our data was in Bengali, we used BNLTK, an open-source python package for Natural Language Processing in Bengali. For our preprocessing we used BNLTK for tokenization, stemming etc. along with Scikit-learn and some regex operations to prepare our data and make it more suitable for the experiments.

Other most used packages include Tensorflow and Keras for deep learning models, Imbalanced-learn to handle class imbalance and some other model specific packages.
We also used Matplotlib and Seaborn to visualize and evaluate how the models were being trained and how they performed.

4.2 Testing and Comparative Analysis

4.2.1 Evaluation Metrics

We used some standard classification metrics to evaluate the performance of our models. These include:

- **Accuracy**: Overall accuracy of the model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision**: The ratio of true positives among those that were predicted as positives.

$$Precision = \frac{TP}{TP + FP}$$

- **Recall**: The ratio of true positives among all actual positives.

$$Recall = \frac{TP}{TP + FN}$$

- **F1-score**: The harmonic means of precision and recall.

$$Accuracy = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

These metrics determined for us how each model performed in detecting the topics and

sentiments from the Bengali comments.

4.2.2 Model performance

After testing simple, deep learning and hybrid machine learning models we have achieved performance ranging from bad, okay to very good with different models (table 4.1,4.2). Here are two comparative tables showing the performance of each model.

Table 4.1: Model performance on Topic detection

Model	Accuracy	Precision	Recall	f1-score
SVM	0.79	0.8	0.79	0.79
CNN	0.93	0.92	0.93	0.93
LSTM	0.95	0.95	0.95	0.95
BiLSTM	0.95	0.94	0.95	0.94
Banglabert	0.79	0.71	0.7	0.71
Multinomial Naive Bayes	0.75	0.74	0.75	0.73
LogisticRegression	0.7	0.7	0.7	0.7
RandomForestClassifier	0.87	0.88	0.88	0.88
XGBoost	0.79	0.8	0.79	0.8
CNN-BiLstm-SVM stack	0.96	0.96	0.96	0.96
CNN-RNN-LR	0.66	0.67	0.66	0.65

Table 4.2: Model performance on Sentiment detection

Model	Accuracy	Precision	Recall	f1-score
SVM	0.81	0.82	0.8	0.81
CNN	0.92	0.91	0.92	0.91
LSTM	0.92	0.92	0.92	0.92
BiLSTM	0.93	0.93	0.93	0.93
Banglabert	0.75	0.73	0.75	0.74
Multinomial Naive Bayes	0.76	0.83	0.71	0.75
LogisticRegression	0.68	0.67	0.7	0.68
RandomForestClassifier	0.89	0.9	0.87	0.88
XGBoost	0.77	0.83	0.72	0.76
CNN-BiLstm-SVM stack	0.93	0.92	0.93	0.92
CNN-RNN-LR	0.64	0.64	0.64	0.63

4.3 Results and Discussion

The traditional machine learning models did not perform very well in our case. Even though some achieved relatively better accuracy than the others, they still pale in comparison compared to the deep learning models. None of the traditional models crossed 90% accuracy but all of the deep learning models and their hybrid easily crossed that margin. The confusion matrix for every model along with their loss curve is given below for better insight.

SVM:

The topic model accurately predicts most topics but struggles with the 'others' category, while the sentiment model excels at identifying 'negative' and 'positive' sentiments but confuses 'neutral' with negative and hate as seen in figure 4.1.

Figure 4.1: SVM confusion matrix

CNN:

Like SVM this topic model also struggles with the 'others' topic and the sentiment model makes some minor misclassification between neutral and negative (figure 4.2). Both the sentiment and topic models show a sign of overfitting while training loss (figure 4.3) continues to decrease.

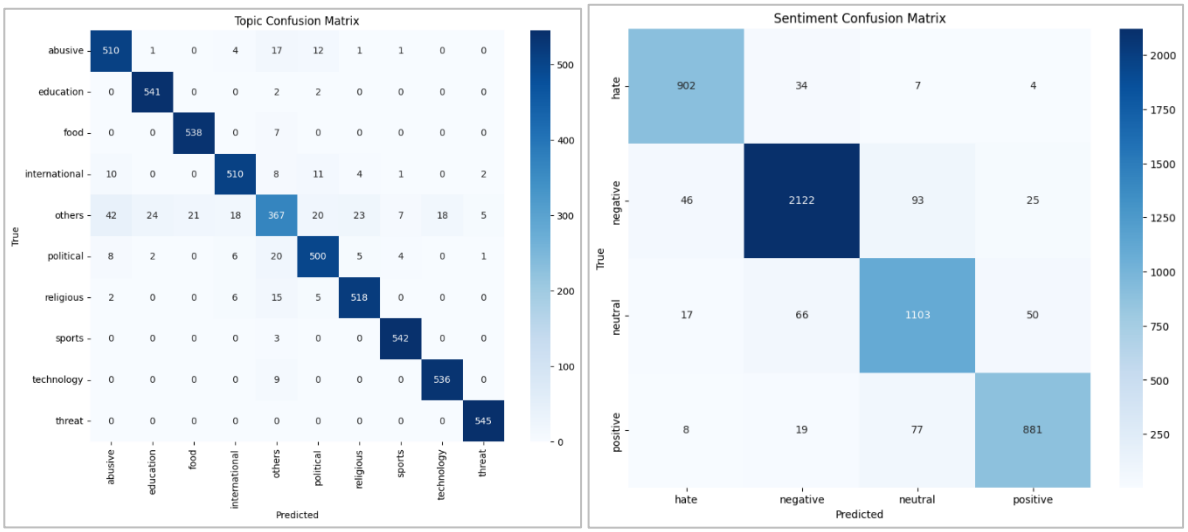

Figure 4.2: CNN confusion matrix

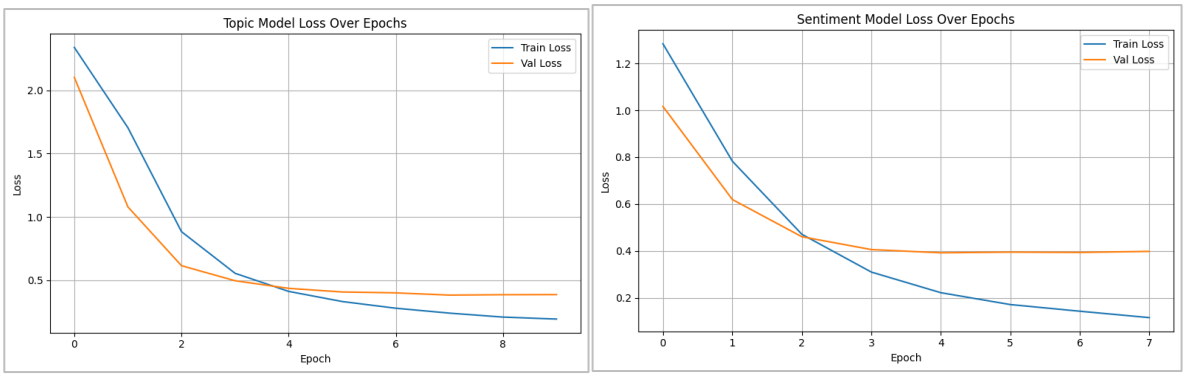

Figure 4.3: CNN loss curve

LSTM:

The sentiment model is highly accurate here but the topic model still struggling with the 'others' category (figure 4.4). Both models show signs of overfitting. The validation loss for topic model stays the same (figure 4.5) while for the sentiment model it keeps increasing slightly.

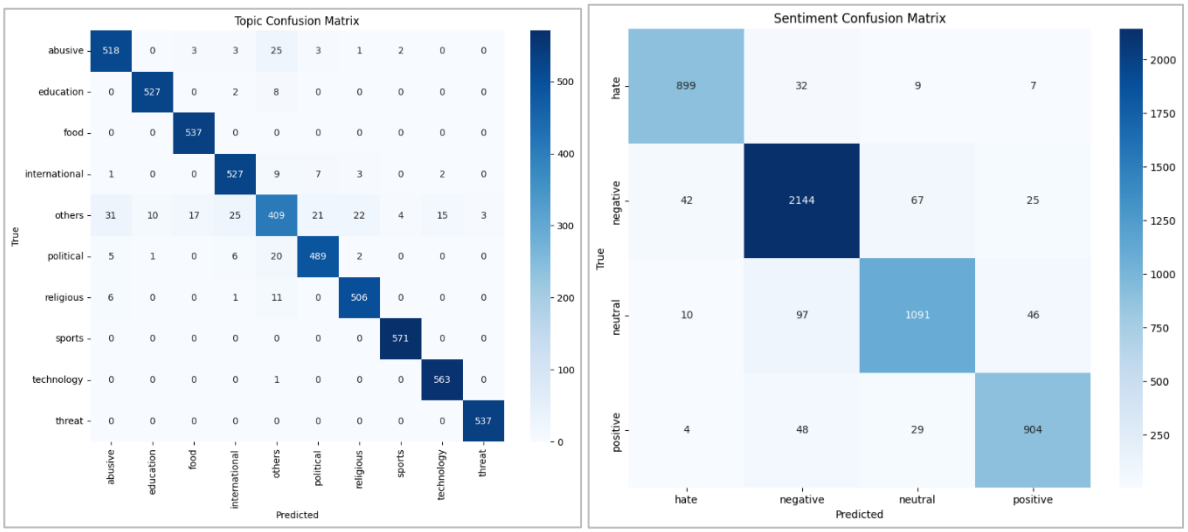

Figure 4.4: LSTM confusion matrix

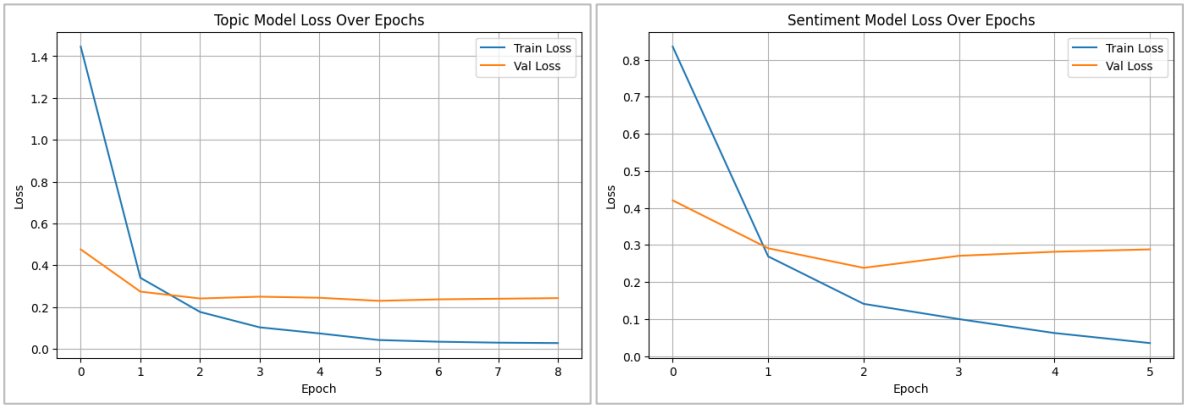

Figure 4.5: LSTM loss curve

BiLSTM:

The sentiment model confuses 'neutral' with 'negative' labels, while the topic model struggles with the 'others' category (figure 4.6). The loss curve (figure 4.7) is similar to the LSTM model. The validation loss is level for the topic model while it keeps increasing for the sentiment model.

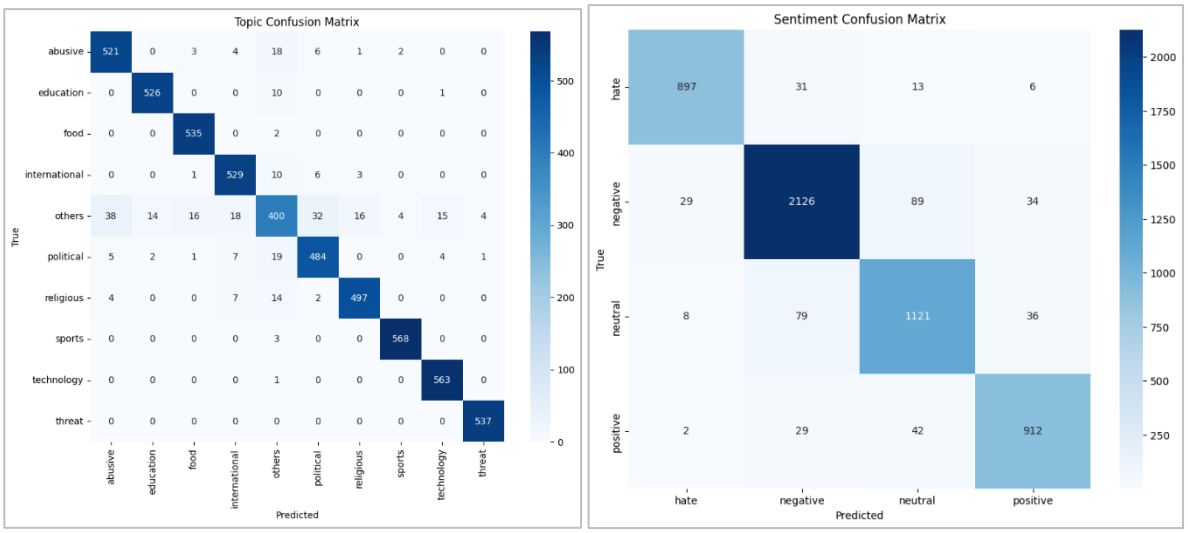

Figure 4.6: BiLSTM confusion matrix

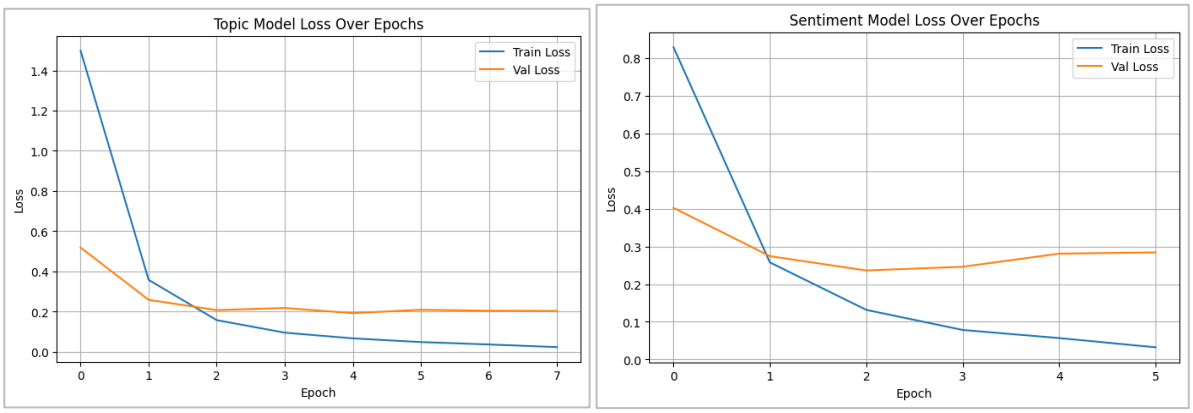

Figure 4.7: BiLSTM loss curve

Bangla-BERT:

The sentiment model here struggles most with classifying 'neutral' comments, which are often predicted as negative. The topic model has low accuracy on the 'others', technology' and 'threat' categories (figure 4.8). Both the topic and sentiment models show a consistent but noisy decrease in training loss (figure 4.9) over time.

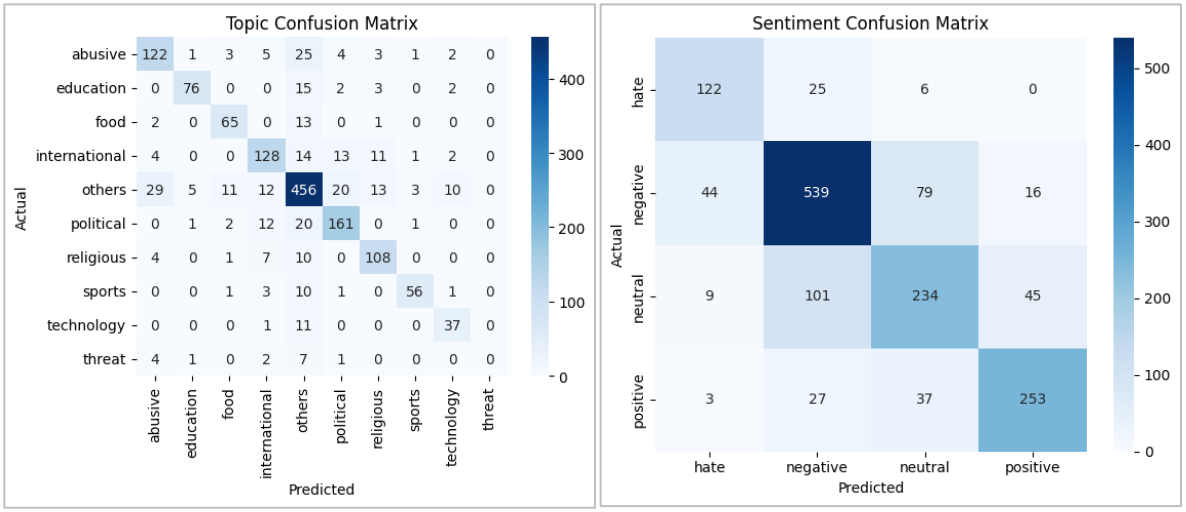

Figure 4.8: Bangla-BERT confusion matrix

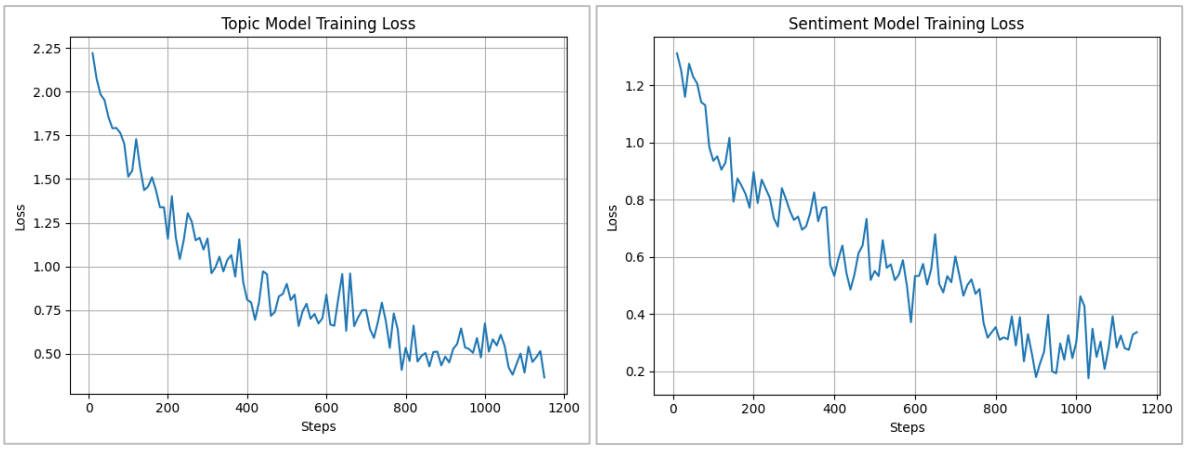

Figure 4.9: Bangla-BERT loss curve

Multinomial Naive Bayes:

The topic model performs well for most categories but misclassifies the 'others' and 'political' topics sometimes (figure 4.10). The sentiment model shows confusion between 'hate' and 'negative' labels. Both the topic and sentiment models show an increase in both training and validation accuracy (figure 4.11) over time.

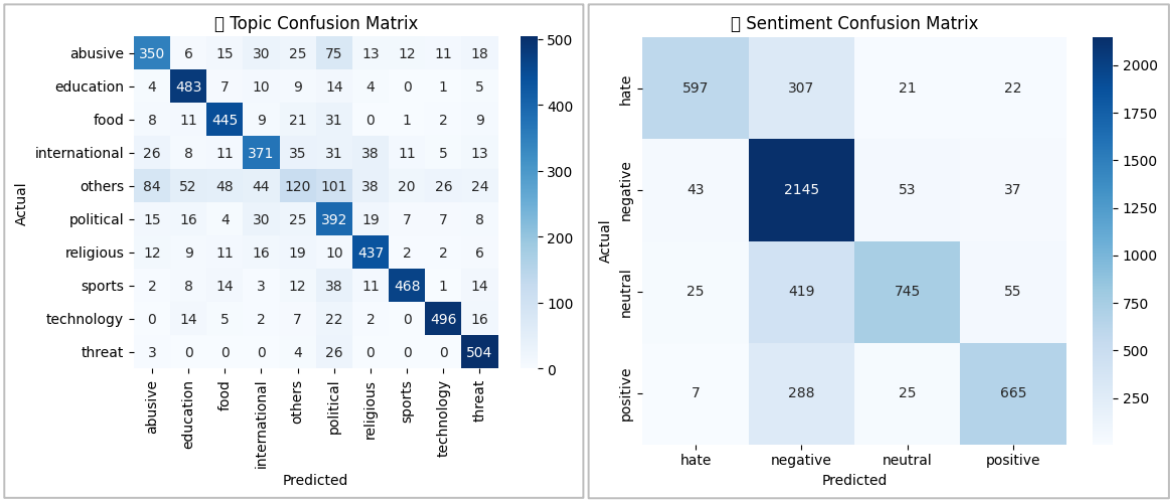

Figure 4.10: Multinomial Naïve Bayes confusion matrix

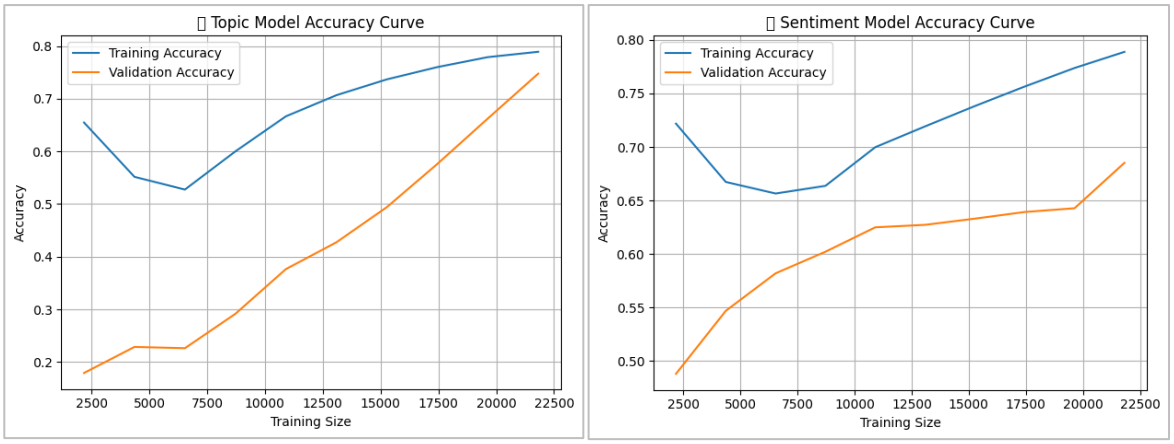

Figure 4.11: Multinomial Naïve Bayes loss curve

Logistic Regression:

The topic model in this case shows widespread misclassification especially with 'others' and 'abusive' categories. The sentiment model is mostly accurate with some misclassifications with 'neutral' and 'negative' categories (figure 4.12).

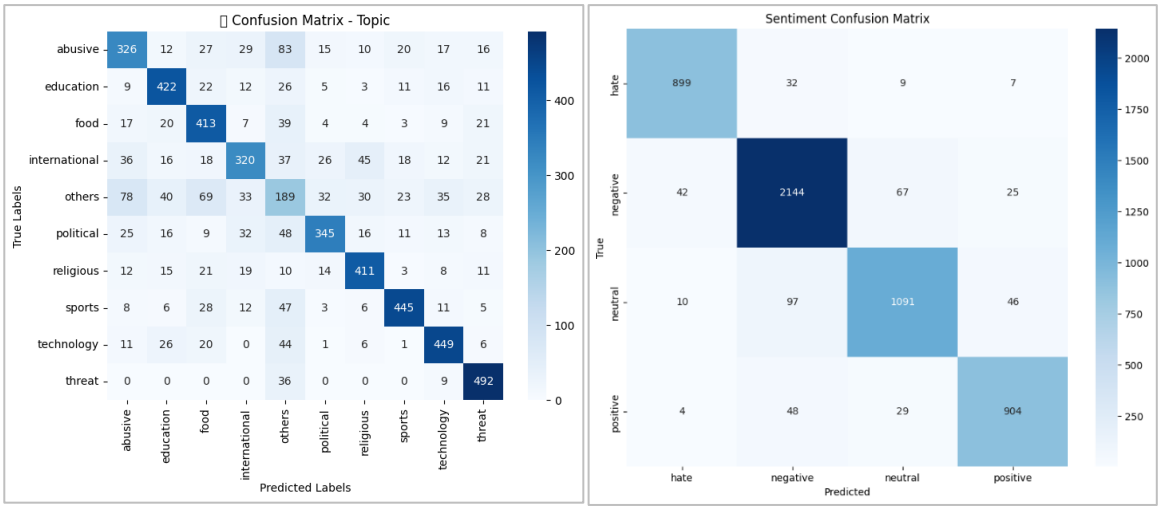

Figure 4.12: Logistic Regression confusion matrix

Random Forest Classifier:

The sentiment model here is accurate but shows confusion between hate and negative labels. The topic model struggles with the 'others' category and often misclassifies abusive comments as 'others' (figure 4.13).

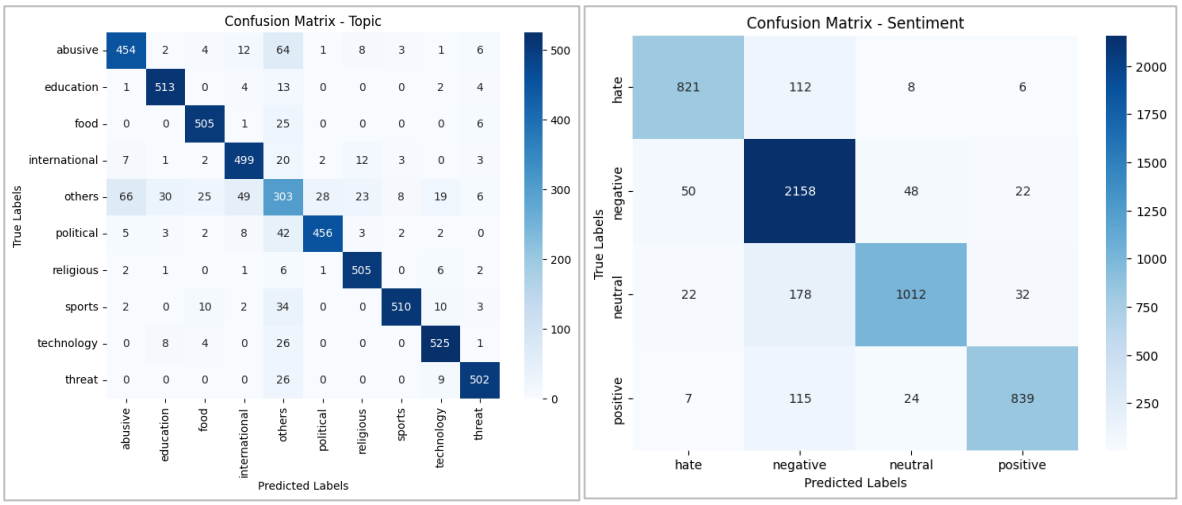

Figure 4.13: Random Forest Classifier confusion matrix

XGBoost:

The sentiment model here confuses 'hate' with 'negative' labels and the topic model struggles with the 'others" category (figure 4.14).

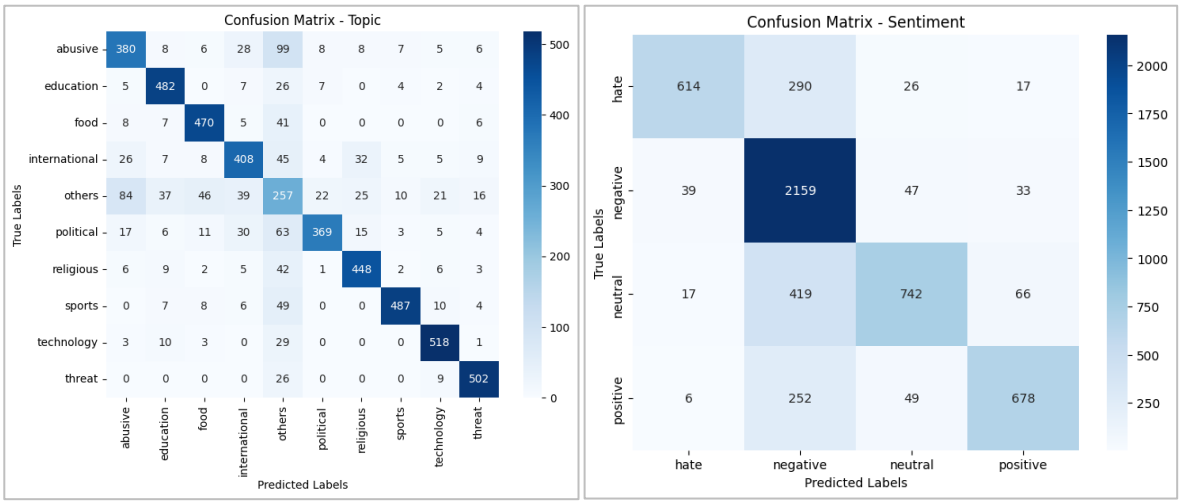

Figure 4.14: XGBoost confusion matrix

CNN-BiLSTM SVM stacking:

This method shows better results than the previous ones. Both the sentiment and topic models perform very well. The sentiment model shows minor confusion between 'neutral' and 'negative' sentiments (figure 4.15) and the topic model's only significant weakness is the 'others' category. The sentiment and topic models show high precision and recall and AUC scores (figure 4.16, 4.17). Only the 'others' category shows weaker than average performance for the topic model.

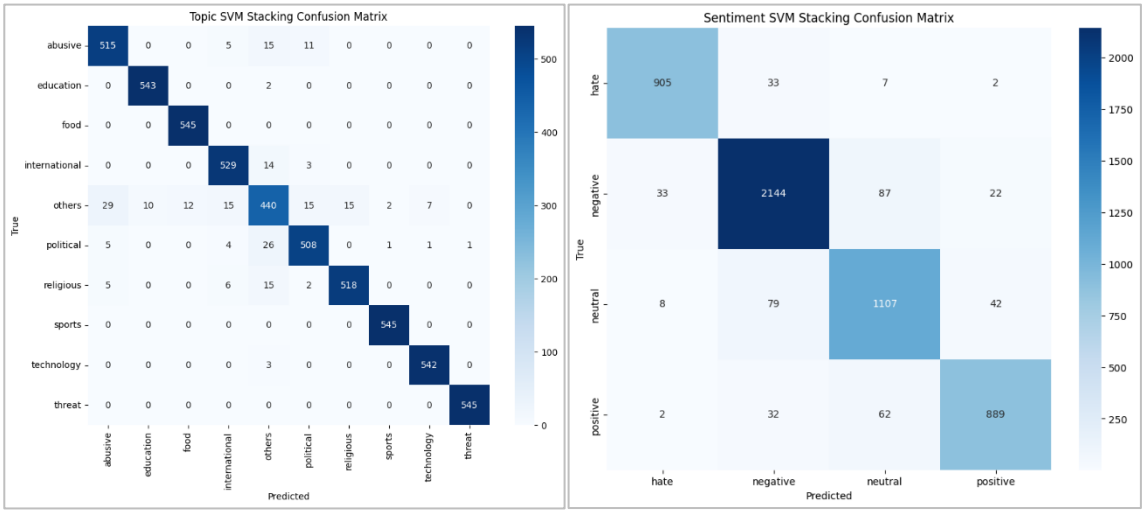

Figure 4.15: CNN-BiLSTM SVM stacking confusion matrix

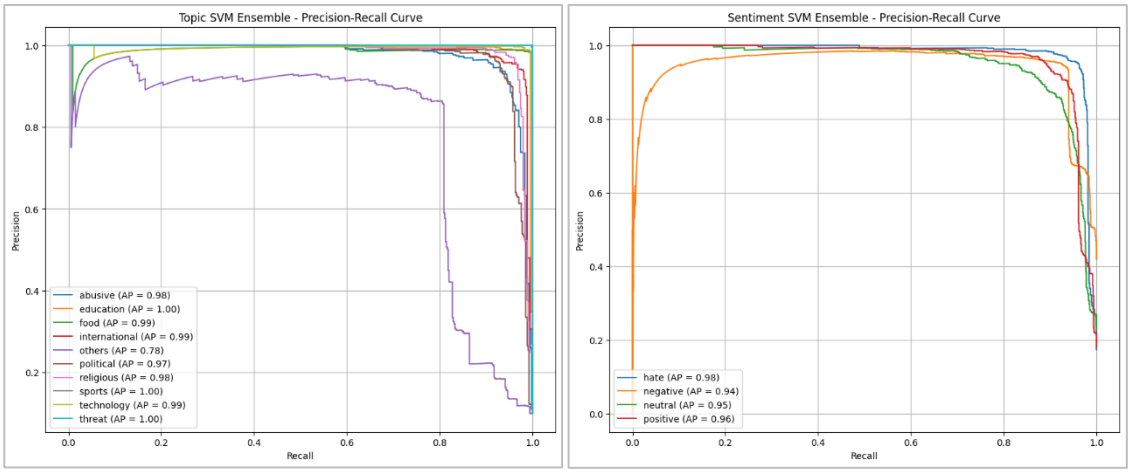

Figure 4.16: CNN-BiLSTM SVM stacking precision curve

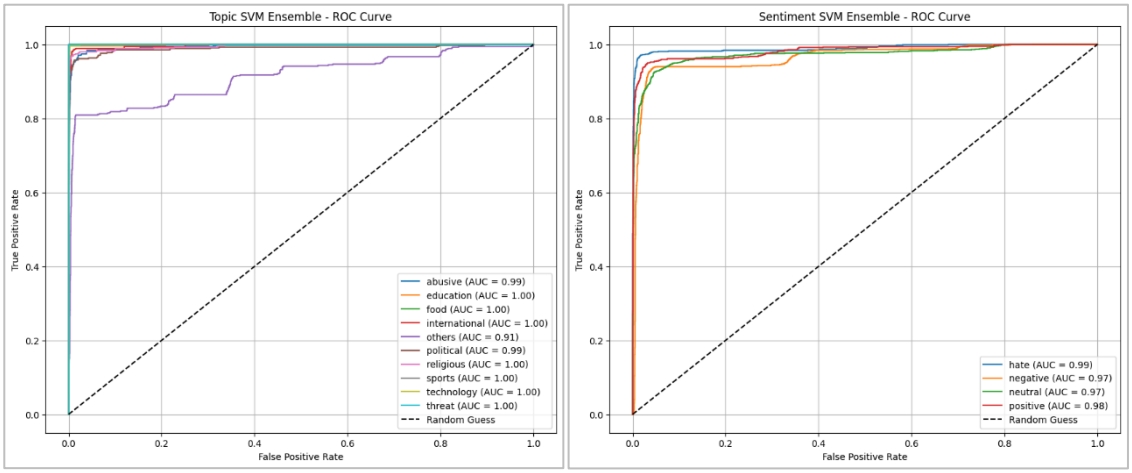

Figure 4.17: CNN-BiLSTM SVM stacking ROC curve

CNN-RNN-Logistic Regression hard voting:

In this method we used CNN, RNN and Logistic regression for prediction. Then we used hard voting to get the final prediction. The result is shown in figure 4.18.

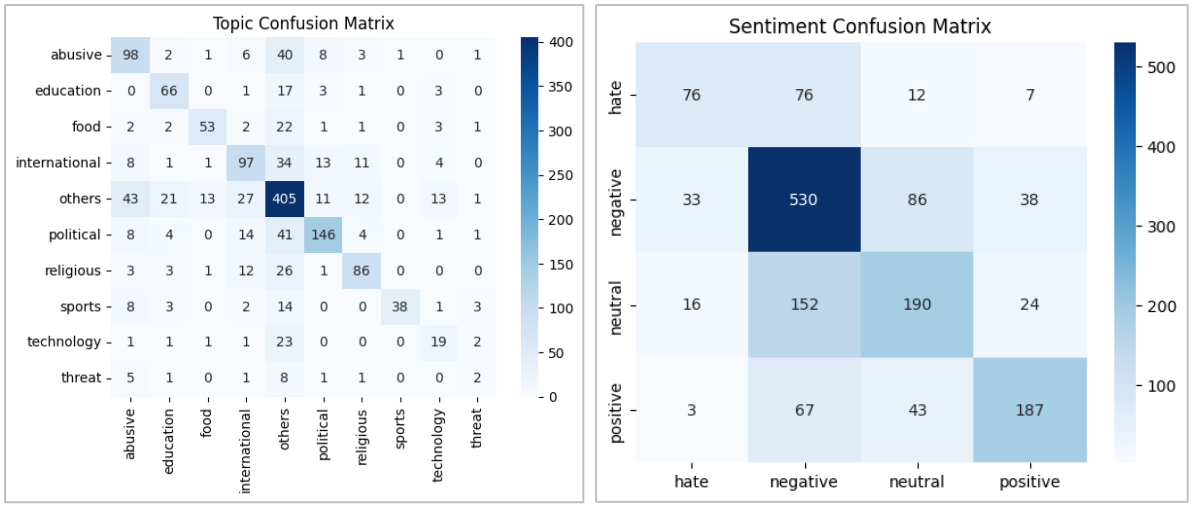

Figure 4.18: CNN-RNN-Logistic Regression confusion matrix

4.4 Summary

The results we received from our experiment with every model clearly indicate that the traditional machine learning models such as Logistic Regression, Naive Bayes and Random Forest clearly get outperformed by deep learning approaches. Especially our proposed CNN-BiLSTM hybrid model achieved the highest performance. It predicted 96% of the topics and 93% of the sentiments correctly. This confirms that combining deep learning models is very effective and it can outperform individual models. Overall, the results validate that CNN-BiLSTM-SVM is a powerful and reliable approach for classifying topics and sentiments from Bengali comments. Therefore, it can be used for detecting and filtering toxicity and vulgarity in online platforms and making the internet a safer space for everyone.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with Standards

For our experiments we followed the standards that we considered best for our case. These standards can be divided into multiple categories.

5.1.1 Software Standards

We followed widely adopted machine learning and deep learning standards for our project. We used Python for developing models as it is the most used programming language for AI and machine learning and it has a wide range of libraries that were very necessary and useful for developing our models. Following these standards ensured that our approaches are comprehensive and compatible with existing research.

For alternatives, we could use PyTorch and Scikit-learn, which are also widely used, but we chose to go with Tensorflow and Keras instead because they are more suitable for our use case.

5.1.2 Hardware Standards

We used Jupyter notebook by Google Colab with its T4 GPU provided for free. We used this platform to develop, save and test our models. It provided us with decent training speed and an efficient environment.

Using local hardware is also an option but training deep learning models can require a lot of computational resources which can be very expensive. It is also a hassle to set up an environment appropriate for this purpose. Using Colab was objectively the better choice here by far.

5.1.3 Data Standards

Our data set consists of Bengali comments from various online social platforms. They were collected excluding any data that can be used to identify the users whose comments we collected. The preprocessing of the data was done using common conventions like tokenization, stemming, filtering irrelevant characters etc. A popular library named 'bnltk' was used to perform most of these operations.

We could use the commonly used preprocessing libraries but we chose bnltk and custom-made functions instead because they are more tailored towards preprocessing Bengali text for our model training.

5.2 Impact on Society, Environment and Sustainability

Data analysis has become a necessity in our day-to-day life. Analyzing Bengali comments and using its results for various purpose can have a significant impact on online activity in the Bengali community.

5.2.1 Impact on Life

Our developed models can identify and filter toxic and hateful Bengali comments in online platforms. This can ensure nicer mental health for users. It can protect children and other vulnerable groups of people from online toxicity without active monitoring. This will enable a safer online environment for everyone. The models can be used for other purposes as well. We can use it to analyze the sentiment of a post or a video by its comments like we demonstrated. This can be utilized in various ways for many different purposes.

5.2.2 Impact on Society

In society it's important to maintain respectful and safe interaction between everyone. Automated toxicity detection will discourage toxic users to act this way as their inputs will be filtered out automatically if it does not meet the standards of the model. This way online platforms become a safer space and more trustworthy for its users. Therefore, decreasing the rate of inappropriate behavior in society as a huge portion of the interaction between people takes place online.

5.2.3 Ethical Aspects

Developing our models requires us to make many choices that can lead to unethical actions if we make the wrong decisions. Making the right choices was of utmost importance in every step we took such as:

- Excluding personal information like names or usernames while collecting data to make the data anonymous
- Using google colab resources only as much as necessary to avoid wasting computational resources
- Only used official API and manual data collection instead of trying to illegally crawl online social platforms.

5.2.4 Sustainability Plan

The models are trained using a sustainable method. The preprocessing, training and usage of the models are all done using conventional reproducible approaches. To update the models all we need to do is train the models using updated datasets using the same approach. This can also make online platforms more sustainable as it can ensure a safer environment for its users. Lastly, if anyone wants to improve or experiment with the models, they can easily do that since everything about our research is well documented for this purpose.

5.3 Project Management and Financial Analysis

Managing the project required meticulous planning and execution. We had to make sure that we maintained all required standards while producing optimal results while staying

within our financial limits.

5.3.1 Project Management

To ensure proper execution of our project we divided it into several phases. These phases are:

- **Problem definition:** Identified that there was a need for the ability to classify topics and sentiments from Bengali text which can be used for various purposes including detecting toxicity on online platforms.
- **Data collection:** Collected data from various online platforms such as Facebook, Twitter, YouTube etc. and applied preprocessing on them.
- **Model training:** We trained and tested several traditional and deep learning models and explored newer approaches by creating hybrid models.
- **Model evaluation:** We used standard performance metrics such as accuracy, f1-score, precision, recall to evaluate the models. Then compared the trade-offs between the simple and complex models.
- **Documentation:** Prepared a structured document of every phase of our project in a standard and comprehensive manner.

5.3.2 Financial Analysis

Since our project was experimental and we used Google Colab platform for training our models, our costs were minimal. But an estimate cost for a stable working version of our project hosted on a scalable platform is given below:

Table 5.1: Estimated cost

Category	Description	Monthly Cost (BDT)
Model Training (GPU)	AWS EC2 g4dn.xlarge (NVIDIA T4 GPU) – On-Demand price for ~40 hrs/month	2,520
API Hosting (CPU VM)	AWS EC2 t3.medium (2 vCPUs, 4 GB RAM) – 24/7 inference	3,640
Storage	AWS S3 Standard – 20 GB data + models	55
Data Transfer (Bandwidth)	AWS outbound traffic – 10 GB/month	108
Monitoring & Logging	AWS CloudWatch (basic metrics + logs)	182
Domain & SSL	Domain registration + SSL certificate (annualized monthly)	1,000
Maintenance (Support)	Developer effort (~20 hrs/month at 1,800 BDT/hr)	36,000

Estimated Monthly Total: ≈ 44000BDT

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

Table 5.2: Mapping with Complex Engineering Problem.

EP1 Dept of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stake-holder Involvement	EP7 Interdependence
✓	✓	✓	✓	✓		✓

The table 5.2 above lists seven criteria for complex problem solving. Each check marks indicates which criteria apply to our project.

EP1: Depth of Knowledge
This refers to technical knowledge required to implement the project. In our case, the technical knowledge for preprocessing, model building is required.

EP2: Range of Conflicting Requirements
Many engineering problems include conflicting goals, where the engineer has to choose which to keep while sacrificing another. Since our case handles both the model and website, we have to decide whether we want a robust model with slow reply or a fast less robust model. It's a speed vs accuracy decision.

EP3: Depth of Analysis
This refers to breaking complex problem to small parts and applying analysis and testing before conclusion. The project involves testing with multiple models before conclusion.

EP4: Familiarity of Issues
Some problems are well researched or has some base research done while others require novel approach. The .machine learning models are well researched but needs adaptation for local Bengali language

EP5: Extent of Applicable Codes
Engineering projects require abiding by set rules and codes. For the data collection for the project user anonymity was kept according to the ethical codes and google terms of service.

EP6: Extent of Stakeholder Involvement
This refers to the stakeholder in the projects. The project does not have any external stakeholders in place.

EP7P: Interdependence

Interdependence means problem components are interconnected , where one depends on another. In the project the result of preprocessing directly affects the model training.

Mapping with Knowledge Profile

Table 5.3: Mapping with knowledge Profile.

K1 Natural Science	K2 Mathematics	K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K7 Comprehension	K8 Research Literature
	✓	✓	✓	✓	✓	✓	✓

The table 5.3 above lists eight criteria for Mapping knowledge Profile. Each check marks indicates which criteria apply to our project.

K1: Natural Science

This indicates applying concepts from physics, chemistry, biology or other related sciences to engineering problems. Our project does not require any natural science knowledge.

K2: Mathematics

Knowledge in Mathematics forms is necessary to understand the concepts of modeling, algorithms, probability and optimization of our project because it's heavily reliant on machine learning and deep learning algorithm. These algorithms require mathematical concepts to fully understand them.

K3: Engineering Fundamentals

Engineering fundamentals are the core engineering disciplines such as mechanics, circuits, materials etc. which are essential for traditional engineering fields. Our project is software-based and it's necessary to have Engineering fundamentals knowledge for this.

K4: Specialist Knowledge

This means specializing in a certain domain that can help solve domain-specific problems for our project. Since this project requires both machine learning and some web development knowledge, Specialized knowledge is required for us.

K5: Engineering Design

This refers to the structured process of designing systems to meet specific requirements. In our case we trained models in a systematic way. Then we created a web application that also requires designing a proper flow of functionality.

K6: Engineering Practice

This means the practical application of engineering principles in real-world scenarios. Since we created a web application using our trained models, it's a practical and real-world implementation.

K7: Comprehension

Comprehension is the ability to understand, interpret problems, methods and solutions. This is required in our case since comprehension was needed both during experimentation and implementation of the web application.

K8: Research Literature

This indicates researching and utilizing the already existing academic and industry work to avoid duplicate work and taking the previous works to the next level. Since we relied on multiple previous research to guide our model training, it's required for our project.

5.4.2 Engineering Activities

Mapping with Complex Engineering Activities

Table 5.4: Mapping with Complex Engineering Activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓		✓	✓	✓

The table 5.4 above lists five criteria for Mapping Complex Engineering Activities. Each check marks indicates which criteria apply to our project.

EA1: Range of Resources

Project may require multiple resources like software, hardware, human experts etc. This project involves usage of google colab gpu, various machine learning libraries and google API,

EA2: Level of Interaction

Some project requires extensive external communication. This project is self-sustained with minimal interaction required. Most of the interaction happen internally while calculation.

EA3: Innovation

This refers to applying different methods, combination and techniques other than traditional ones. This project encompasses novel approaches with new dataset, model building and application.

EA4: Consequences for Society and Environment

Engineering problems can benefit both society and environment. Although the project has little effect in environment, it can help society by better understanding the human behavior.

EA5: Familiarity

IT refers to the familiarity of the subject. NLP and AI methods are well known and familiar, however it requires modification and adaptation for Bengali language.

5.5 Summary

Our project develops a system for Bengali comment classification and toxicity detection using machine learning and deep learning approaches. We followed widely used platforms, software and data standards, including Python, TensorFlow/Keras and UTF-8 text encoding for Bengali characters in a GPU enabled environment provided by Google Colab.

Key challenges included collecting data from various sources and handling them because the available programming resources for a language like Bengali is not that abundant. To overcome the results achieved by the traditional deep learning models we had to try a newer approach to combine multiple models to produce a hybrid approach that managed to achieve better results. Ethical aspects such as data privacy and impact on life and society were also considered. Overall, the project uses standard software practices with practical solutions for Bengali NLP and produces a system that can classify topics and sentiments from Bengali text and use the results for relevant purposes.

Chapter 6

Conclusion

6.1 Summary

This report studies the usage of hybrid deep learning models in informal social media comments and identifies cyber crimes and how it can be used to make the cyberspace more friendly. Since informal language differs from regular writings and contains much more features machine learning models often fails to accurately classify them. The study found that most traditional models are bound to 70-80% accuracy. To solve this Deep learning models were used. This proved much more effective, obtaining an accuracy upto 94%. To further strengthen it multiple deep learning models were used to obtain the highest of 96% accuracy. The evaluation was done by classification report, confusion matrix and validation loss. AUC and ROC curve were also plotted to understand the models behavior and avoid overfitting. This study contributes a corpus of 7725 labelled comments which can be used to further enhance Bengali models. It also provides a website for analytic usage. These along with the hybrid models will help social media platform to automate cyber crime monitoring and analyze post feedback.

6.2 Limitation

Limitation of this study is in the data availability and resource needed for labelling a total of 14 classes dataset. A large number of dataset would help make it a full fledged all round model ready for deployment. Another limitation is lack of variety in collection platforms. Since most platform don't allow web scrapping its limited to a few platforms and some in manual collection which is time consuming. This study also proved poor result in machine learning models which can be improved for better comparison. The Hybrid deep learning models are also limited to fixed hyper parameter which can be improved . It also requires long time to analyze entire comment section. The created website is not hosted in public space due to a lack of fund. It can be further improved for more visual appeal and traffic handling. All of these limitations hint at the improvement option for future.

6.3 Future Work

There's scope to use this study to improve future work.

- The contributed dataset is unique and contains more classes than other contribution which will help further improve future models.

- The model can be further improved and used on larger dataset to improve

accuracy and classification of unknown words.

- Using Hyperparameter tuning will improve the accuracy and reliance on fixed input.

- This study has the potential to be used by social media platforms to automate hate crime and cyber bullying detection, making it a safer place for everyone.

- The website can be used a base to create a commercial website ready to use for media analysts to gather feedback by topic.

The scope of this study in future works are endless. This study have created and contributed a base for that.

References

[1] Das, A. K., Al Asif, A., Paul, A., & Hossain, M. N. (2021). Bangla hate speech detection on social media using attention-based recurrent neural network. Journal of Intelligent Systems, 30(1), 578-591.

[2] Karim, M. R., Dey, S. K., Islam, T., Sarker, S., Menon, M. H., Hossain, K., ... & Decker, S. (2021, October). Deephateexplainer: Explainable hate speech detection in under-resourced bengali language. In 2021 IEEE 8th international conference on data science and advanced analytics (DSAA) (pp. 1-10). IEEE.

[3] Rafin, Rayhan; Shibly, Mohammad Sohaib Islam (2025), "Bangla Social media comments Dataset (BanglaMedia)", Mendeley Data, V1, doi: 10.17632/xyxb5kryx3.1Haque, R., Islam, N., Tasneem, M., & Das, A. K. (2023). Multi-class sentiment classification on Bengali social media comments using machine learning. International journal of cognitive computing in engineering, 4, 21-35.

[4] Romim, N., Ahmed, M., Talukder, H., & Saiful Islam, M. (2021, May). Hate speech detection in the bengali language: A dataset and its baseline evaluation. In Proceedings of International Joint Conference on Advances in Computational Intelligence: IJCACI 2020 (pp. 457-468). Singapore: Springer Singapore.

[5] Awal, M. A., Rahman, M. S., & Rabbi, J. (2018, October). Detecting abusive comments in discussion threads using naïve bayes. In 2018 international conference on innovations in science, engineering and technology (ICISET) (pp. 163-167). IEEE.

[6] Banik, N., & Rahman, M. H. H. (2019, December). Toxicity detection on bengali social media comments using supervised models. In 2019 2nd international conference on Innovation in Engineering and Technology (ICIET) (pp. 1-5). IEEE.

[7] Chakraborty, P., & Seddiqui, M. H. (2019, May). Threat and abusive language detection on social media in bengali language. In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT) (pp. 1-6). IEEE.

[8] Emon, E. A., Rahman, S., Banarjee, J., Das, A. K., & Mittra, T. (2019, June). A deep learning approach to detect abusive bengali text. In 2019 7th International Conference on Smart Computing & Communications (ICSCC) (pp. 1-5). IEEE.

[9] Ishmam, A. M., & Sharmin, S. (2019, December). Hateful speech detection in public facebook pages for the bengali language. In 2019 18th IEEE international conference on machine learning and applications (ICMLA) (pp. 555-560). IEEE.

[10] Asghar, M. Z., Subhan, F., Ahmad, H., Khan, W. Z., Hakak, S., Gadekallu, T. R., & Alazab, M. (2021). Senti-eSystem: a sentiment-based eSystem-using hybridized fuzzy

and deep neural network for measuring customer satisfaction. Software: Practice and Experience, 51(3), 571-594.

[11] Akhter, S. (2018, December). Social media bullying detection using machine learning on Bangla text. In 2018 10th International Conference on Electrical and Computer Engineering (ICECE) (pp. 385-388). IEEE.

[12] Wahid, M. F., Hasan, M. J., & Alom, M. S. (2019, September). Cricket sentiment analysis from bangla text using recurrent neural network with long short term memory model. In 2019 International Conference on Bangla Speech and Language Processing (ICBSLP) (pp. 1-4). IEEE.

[13] Drovo, M. D., Chowdhury, M., Uday, S. I., & Das, A. K. (2019, June). Named entity recognition in Bengali text using merged hidden Markov model and rule base approach. In 2019 7th International Conference on Smart Computing & Communications (ICSCC) (pp. 1-5). IEEE.

[14] Karim, M. A., Kaykobad, M., & Murshed, M. (Eds.). (2013). Technical challenges and design issues in bangla language processing. IGI Global. Rahman, M., Talukder, M. R. A., Setu, L. A., & Das, A. K. (2022). A dynamic strategy for classifying sentiment from Bengali text by utilizing Word2vector model. Journal of Information Technology Research (JITR), 15(1), 1-17.

[15] Khan, M. S. S., Rafa, S. R., & Das, A. K. (2021). Sentiment analysis on bengali facebook comments to predict fan's emotions towards a celebrity. Journal of Engineering Advancements, 2(03), 118-124.

[16] Kamyab, M., Liu, G., & Adjeisah, M. (2021). Attention-based CNN and Bi-LSTM model based on TF-IDF and glove word embedding for sentiment analysis. Applied Sciences, 11(23), 11255.

[17] Bhowmik, N. R., Arifuzzaman, M., Mondal, M. R. H., & Islam, M. S. (2021). Bangla text sentiment analysis using supervised machine learning with extended lexicon dictionary. Natural Language Processing Research, 1(3), 34-45.

[18] Prottasha, N. J., Sami, A. A., Kowsher, M., Murad, S. A., Bairagi, A. K., Masud, M., & Baz, M. (2022). Transfer learning for sentiment analysis using BERT based supervised fine-tuning. Sensors, 22(11), 4157.

[19] Haque, R., Islam, N., Islam, M., & Ahsan, M. M. (2022). A comparative analysis on suicidal ideation detection using NLP, machine, and deep learning. Technologies, 10(3), 57.

[20] Chowdhury, R. R., Hossain, M. S., Hossain, S., & Andersson, K. (2019, September). Analyzing sentiment of movie reviews in bangla by applying machine learning techniques. In 2019 international conference on bangla speech and language processing (ICBSLP) (pp. 1-6). IEEE.

213-15-4278

ORIGINALITY REPORT

24%	18%	17%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	7%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	Submitted to United International University Student Paper	1%
4	peerj.com Internet Source	1%
5	Rezaul Haque, Naimul Islam, Mayisha Tasneem, Amit Kumar Das. "MULTI-CLASS SENTIMENT CLASSIFICATION ON BENGALI SOCIAL MEDIA COMMENTS USING MACHINE LEARNING", International Journal of Cognitive Computing in Engineering, 2023 Publication	1%
6	www.preprints.org Internet Source	1%
7	aclanthology.org Internet Source	1%
8	www.lrec-conf.org Internet Source	1%
9	Submitted to University of Finance – Marketing Student Paper	<1%
10	Submitted to Coventry University Student Paper	<1%