Advancing Sentiment Classification in Bangla Text: An Enhanced BERT Approach on the SentNoB Dataset

by

Ahmed Wasi Bin Faruque
20101352
Naila Gani
20101351
Maisha Binte Monowar
20101350
Emam Hasan
20301263
Kashfiquzzaman Ratul
20101370

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

Department of Computer Science and Engineering
Brac University
October 2024

Declaration

It is hereby declared that

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

Student's Full Name & Signature:

Ahmed Wasi Bin Faruque
20101352

Naila Gani
20101351

Maisha Binte Monowar
20101350

Emam Hasan
20301263

Kashfiquzzaman Ratul
20101370

Approval

The thesis/project titled "Advancing Sentiment Classification in Bangla Text: An Enhanced BERT Approach on the SentNoB Dataset" submitted by:

1. Ahmed Wasi Bin Faruque (20101352)
2. Naila Gani (20101351)
3. Maisha Binte Monowar (20101350)
4. Emam Hasan (20301263)
5. Kashfiquzzaman Ratul (20101370)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 22, 2024.

Examining Committee:

Supervisor (Member):

Mr. Arif Shakil
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor (Member):

Dr. Farig Yousuf Sadeque
Associate Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department (Chair):

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Since social media has entered into our lives, it has set the scene for unmatched online communication and information sharing for a significant amount of time. These platforms enable people to share their thoughts, opinions and experiences, even in their native languages, which can be used as an asset for sentiment analysis. With this in mind, the paper conducts research about the application of Natural Language Processing (NLP) to evaluate and analyze sentiments from social media posts in Bangla language. In this present world, people tend to share their point of views across social medias over ongoing topics. The ample amount of personalized program data on numerous social media platforms presents an opportunity to gather large scales of information and to use non-invasive tools for sentiment analysis. The following research incorporates insights from twenty relevant studies, providing a clear image of existing methodologies and approaches. This study also acknowledges the challenges and opportunities that come with scooping out information from social media data, including issues of privacy, ethics, and data quality. Moreover, the research utilizes a combination of numerous NLP techniques, sentiment analysis and machine learning algorithms to embody robust models capable of identifying text sentiments accordingly. The proposed methodology's observational assessment involves a large scale of social media databases which allows to assess the performance of models in real world aspects. The discovery illustrates the promising NLP-driven solutions in quick detection and in performing sentiment analysis.This study aims to detect users' sentiment based on their posts posted in Bangla language with the help of BERT (BanglaBERT) and an ensemble algorithm (LSTM + BanglaBERT). This research highlights the potential of NLP-based approaches, specifically utilizing BERT in order to effectively identify and analyze mental health signals from social media posts in Bangla-offering a valuable tool for early intervention and mental health awareness.

Keywords: Machine Learning Algorithms; Natural Language Processing(NLP);Social Media; BERT; Sentiment Analysis.

Dedication

To the brave martyrs of the July Revolution of 2024: your courage and sacrifice have inspired an entire nation, lighting the flames of hope and change. We dedicate this work to your memory. It is because of your strength that we stand here today, with renewed purpose and a clear vision for a better Bangladesh.

To our each team-member: your hard work, creativity, and commitment have been the foundation of this project. This journey would not have been possible without your support, ideas, and friendship. I am deeply grateful to each and every one of you for making this a shared achievement.

To our dear parents: your unwavering belief in us and your constant encouragement have been the pillars of our success. Your love, sacrifices, and guidance have helped us through the toughest times, and for that, we are forever thankful. Thank you for always standing by us with endless patience and faith in our dreams.

This work is as much yours as it is ours.

Acknowledgement

Firstly, all praise to the Almighty Allah - without His mercy, the completion of our thesis without any major interruption would not have been possible. Secondly, to our supervisor Mr. Arif Shakil sir and our co-supervisor Dr. Farig Yousuf Sadeque sir for their kind support and advice in our work. We are truly grateful for all the guidance, help and continuous support by them.

Table of Contents

List of Figures

List of Tables

Nomenclature

Several abbreviations used in the document are listed below:

NLP Natural Language Processing

BERT Bidirectional Encoder Representations from Transformers

SVM Support Vector Machine

XLM-ROBERTa Cross-lingual Language Model - Robustly Optimized BERT Approach

ML Machine Learning

DSM Diagnostic and Statistical Manual of Mental Disorders

LSTM Long Short-Term Memory

Chapter 1

Introduction

Social media platforms have become an integral part of our everyday lives. These platforms have been a fascinating medium for people to share their thoughts and sentiments not only in English but also in their native language. As people are more interested in sharing their sentiments directly or indirectly on social media, these data can be an opportunity to understand people's sentiments on social media using NLP. NLP is a branch of AI that can manipulate and comprehend human language or human written texts. Sentence segmentation, word tokenization, stemming, lemmatization, stopword analysis, dependency parsing and part of speech (POS) tagging - these steps are followed while analyzing the human language. While posting on social media platforms like, Twitter, Facebook, Reddit etc. there are numerous types of formatting or characters used. These informal texts are analyzed by NLP to make it understandable to the machine. Understanding people's sentiments can be done through this. Identifying words used in posts, emojis, sentence structure etc. are observed while analyzing the texts. There could be many direct and indirect use of words that could be signal a user's sentiments. Extracted features can then be used for training machine learning models. The model can then be trained on labeled datasets. This way the model will learn to identify patterns in posts that indicate negative or positive sentiments. One such model is BERT which is based on the structure of BERT model which is specifically designed for Bangla language. Although there have been recent researches done over this topic, most papers were based on English. There are few studies on this topic based on the Bangla language but there is a huge population on social media that expresses themselves in Bangla. This paper tries to explore Bangla language-based posts.

1.1 Problem Statement

Social media platforms have grown significantly as places where people may communicate their thoughts, sentiments and ideas as a result of its rapid expansion. There isn't much study on deciphering users' sentiments from Bangla social media posts in areas like Bangladesh, where Bangla is the primary language. Unfortunately, there is a gap in the accuracy of Bangla text interpretation because the majority of sentiment analysis models now in use were predominantly trained on English datasets. Because of its complex morphology, informal writing styles that are common in social media, and grammatical patterns, the Bangla language possess special issues. The linguistic complexities, combined with the cultural background and restricted availability of annotated datasets, provide challenges to the creation of efficient models for examining Bangla social media communication. This project intends to close this gap by creating an NLP-based model that forecasts users' sentiments using the SentNoB dataset, which is a collection of labelled Bangla social media messages. By utilizing this dataset, we want to improve the accuracy of Bangla sentiment classification and establish a foundation for future studies on social media analysis for sentiment analysis. By offering insights into the difficulties and potential solutions for low-resource languages like Bangla, this research also aims to make a contribution to the larger area of NLP.

1.2 Thesis Structure

Chapter 1 discusses a brief introduction to the idea of the paper along with the problem it is aiming to address and the research objectives. **Chapter 2** consists of a detailed summary of the background studies-researches that were previously published related to our research topic. **Chapter 3** holds a discussion about the datasets used in this work. **Chapter 4** illustrates the proposed methodology in detail, model architecture, training initialization and workflow. **Chapter 5** gives an overview of the whole result analysis process evidenced by classification reports, confusion matrices, training and validation accuracy and loss graphs-finally a total insight on the whole research and overall performances of the models. Finally, **Chapter 6** concludes the research with a summary of the work done in this paper and how it can improve the performance in the future.

1.3 Research Objectives

1. Use BERT and BanglaBERT for sentiment classification on social media posts in Bangla to predict users' sentiment and assess its effectiveness using a labelled dataset such as SentNoB.

2. Using the advantages of both transformer-based and recurrent neural networks, create an ensemble model that combines BERT(BanglaBERT) and LSTM architectures to increase the accuracy of mental state categorisation in Bangla social media postings.

3. When working with low-resource Bangla language data, apply data augmentation techniques—specifically, 'Backtranslation'—to increase the dataset's variety and the machine learning models' resilience.

4. Evaluate how well 'Backtranslation' reduces class disparity and improves the model's ability to generalise to previously undiscovered Bangla social media posts.

5. Contribute to the development of NLP for low-resource languages, Bangla in this case, by showcasing the potential of advanced techniques such as ensemble learning, transformer models and data augmentation to enhance sentiment categorisation.

Chapter 2

Literature Review

To understand different models of NLP and Sentiment Analysis, we have studied some scholarly articles. While selecting articles, we have tried our best to choose articles that are relevant to our field, have a good number of citations, and were published in recent years.

The paper written by Tianlin Zhang, Annika M. Schoene, Shaoxiong Ji & Sophia Ananiadou [1] highlights the effect of mental illness on our health, illustrates the importance of mental health as an individual and society as a whole and finally provides a Natural Language Processing (NLP) based solution for early detection of mental illness. This paper conducts a detailed review which contains 399 studies from the past decade and shows a uprising trend in NLP research in order to detect mental illness; mentioning the types of datasets and NLP methods used, challenges and opportunities of using NLP and the need of having more research in this rapidly evolving field.The paper displays a search and filtering strategy to identify relevant studies. They have searched 6 databases: PubMed, Scopus, Web of Science, DBLP computer science bibliography, IEEE Xplore, and ACM Digital Library - using specific keywords related to mental health and found out 10,476 records; only 7,536 were found unique after removing duplicates. A tool named RobotAnalyst has been used to filter these records based on title and abstract, and it excluded all the non- English and unrelated to mental illness detection articles. The tool efficiently removed all the records using speech or image data and also those not using textual experimental data. After the full filtering action, only 399 studies were included in consideration for the review, including a flowchart and reasons for exclusions of the removed records. From these 399 selected studies, they have extracted data including publication year, dataset, NLP methods, and features used. The review highlights the importance of data quality, accountability and other challenges and opportunities. The paper discusses the different datasets that were gathered from various sources such as social media, surveys, interviews, and electronic health records. These datasets contained diverse mental health conditions such as depression, anxiety, schizophrenia in different languages (e.g., English, Chinese) derived from Twitter, Reddit and clinical records. Mentioning the references to all the NLP methods, the authors outline those methods used such as traditional machine learning methods (SVM, Adaptive Boosting, Decision trees), highlighting the used pipeline approach which includes data pre-processing, optimization, evaluation, feature extraction and modeling. They have also alluded to Convolutional Neural

Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) and other deep learning methods which are used for end to end modeling and feature learning in order to improve the correctness and efficiency of mental illness detection. Linguistic features[Part-of-Speech (POS), Bag-of-Words (BoW), and Linguistic Inquiry and Word Count (LIWC)], statistical features [n-gram, term frequency-inverse document frequency (TF-IDF)], domain knowledge features, and other auxiliary features including social behavioral features, user's profile, and time features are preferred by the authors.The authors discuss the usefulness of NLP for mental illness detection in this paper as well as have mentioned the database names (social media, clinical records, surveys), NLP models used (SVM, Decision Trees, Adaptive Boosting) ; we can try and implement these models in our research as well since we get a brief idea about these databases and models from this discussed paper.

As Islam et al.[2] point out, in addressing significant gaps with regard to resources devoted to Bengali NLP (Natural Language Processing), they have put together a large scale dataset for performing sentiments analysis which focuses on the Bangla language. The SentiGOLD dataset comprises 70,000 samples collected from diverse sources including social media, blogs, and new articles, and is annotated across five sentiment classes. The research has a detailed methodology that includes the use of systematic data gathering and data cleaning techniques, as well as an IAA that is rated at 0.88, so long as the rating is derived from a formal annotation management system and is endorsed by linguistic consultants.This high IAA score reflects the dataset reliability and the careful design of the annotation guidelines. The paper critically evaluates existing Bangla sentiment analysis datasets, highlighting their limitations such as smaller sample sizes, lower IAA scores, and lack of domain diversity. On the other hand, SentiGOLD has a large domain coverage and this is the main reason for it to be a strong resource. The researchers conducted both intra-dataset and cross-dataset evaluations, demonstrating the generalizability of the dataset and competitive performance with the macro F1 score of 0.62 regarding five classes and 0.61 about three classes in tests across various datasets. However, the study acknowledges certain limitations, including challenges with annotating sarcastic and politically sensitive texts, suggesting areas for future improvement. Out of all these systems, SentiGOLD is perhaps the most promising because it could be the first standard for conducting sentiment analysis on the Bangla language, which in turn can encourage further work on low-resource languages. Future work should also seek to expand the dataset by adding more different types of informal text samples to make the model more effective and ready for actual use.

As part of the continuing efforts to improve resources for Bangla Natural Language Processing, Bhattacharjee et al.[3] propose BanglaBERT, which is a BERT based model accommodating the Bangla language which is still lacking considerably in the area.The authors compiled a substantial 27.5 GB corpus from 110 Bangla websites to pretrain the model, dubbed 'Bangla2B+'. BanglaBERT is benchmarked against four NLU (Natural language understanding) tasks: sentiment classification, NER (Named entity recognition), QA (Question answering), and NLI (Natural language inference), through the newly introduced Bangla Language Understanding Benchmark (BLUB). The results indicate that BanglaBERT significantly outperforms existing multilingual models like mBERT and XLM-R, showcasing improvements

in both supervised and zero-shot transfer settings. A notable strength of this research is its comprehensive dataset creation and the meticulous preprocessing steps, including deduplication and noise reduction, ensuring high-quality training data. The implementation of the ELECTRA pretraining method further enhances efficiency, providing robust results with less computation overhead compared to larger multilingual models. However, the study also faces limitations, such as the potential for residual noise in the pretraining data and the ethical implications of web-crawled content. Despite these challenges, The introduction of BLUB along with the initiatives concerning publicly available resources, has played a good role in the Bangla NLP arena, triggering extensive research and development. The paper sets a new standard for Bangla language models, highlighting the need for language-specific resources in low-resource language contexts. Future work could explore extending BLUB to include other NLU tasks and refining pretraining data quality to mitigate any remaining biases of inaccuracies. This study paves the way for advancing computerational resources and tools for Bangla which would further facilitate the cause of linguistic diversity in NLP.

The paper authored by Aziliz Le Glaz, Antoine Messiah, and Antoine Italiano [4] gives a systematic review in mental health research by using Natural Language Processing (NLP) and following PRISMA guidelines. This paper covers 327 articles from 4 databases with 58 included. The target of this study is to identify symptoms, classify illness severity, and offer psychopathological insights. Data sources used are social media alongside medical sources and Python was mainly used to conduct the study. Authors have mentioned the potential of machine Learning(ML) and natural language processing (NLP) in order to improve mental healthcare. The main goal of the paper is to illustrate ML and NLP studies in mental health by focusing on various methods to use in this regard. It involves how NLP emerged in the 1960s and has helped to classify texts, sentiment mining and has performed efficiently in detecting symptoms of several mental health conditions. The contribution of NLP and ML in diagnosing and detecting mental health situations, identifying risk elements, testing treatment adherence and analyzing the side effects are also mentioned here. In this study, preprocessing methods such as part-of-speech tagging,lemmatization,tf-idf, n-grams, and CUI extraction are used in order to prepare a clean textual data. To classify tasks, decision trees, association rules, and neural networks are used; whereas transparent methods like decision trees and association rules are used to have insights into the classification process. Sentiment analysis, emotion analysis, topic detection, and named-entity recognition methods are applied in order to gather more information. From 222 studies gathered from the four databases, only 62 studies were selected on the basis of having relevance to ML and NLP in mental health. The study covers several mental conditions such as addiction, PTSD, neurodevelopmental disorders and also mentions other studies on violence, cyber harassment, treatment and cognitive issues. 3 population categories were studied-patients with EHR, those seen in emergency or psychiatric departments and social media users. The review says studies in mental health research use ML and NLP in order to explore diverse data sources like EHRs and social networks, moreover initiatives such as RDoC, incorporating genetics and neuroscience, challenge traditional diagnostic practices such as DSM. ML and NLP can create homogenous units and identify biomarkers - all these techniques can effectively reshape psychiatric clas-

sifications.The authors have admitted the limitation of having lack of quantitative comparison between the studies and absence of scoring for risk of bias.Hence, from this study, we get a brief idea about data sources; about topics such as psychological insights, data preprocessing methods, classification techniques, symptom detection - we can use these knowledges in our research for data preprocessing, classifying and to acquire more understanding about our topic's field.

Continuing the focus on improving Bangla sentiment analysis, the research by Kazi Toufique Elahi et al.[5] addresses a critical challenge, managing noise in Bangla text sentiment analysis, particularly in social media data. The authors present the NC-SentNoB dataset, which has been carefully annotated to recognize ten different types of noise in around 15,000 noisy Bangla texts. This dataset serves as a basis for evaluating various noise reduction methods, including spell correction, back-translation, paraphrasing, and masked language modeling. The study highlights that pre-trained language models (PLMs) like Bangla-BERT-Base, while effective under controlled conditions, show a marked performance drop (around 50%) when dealing with noisy texts. Experimental results indicate that the current noise reduction methods are insufficient, as models trained on noise-reduced data did not outperform those trained on noisy data, underscoring the necessity for more advanced noise reduction techniques. Strengths of the research include the creation of a detailed noise-annotated dataset and the systematic comparison of multiple noise reduction strategies. However, the study faces some limitations due to the relatively low performance of the noise reduction methods tested and the dataset's imbalance, which could bias the results toward more common types of noise. Since this study includes rather simple noise detection and correction algorithms, future work should focus on how to develop more sophisticated noise reduction models by deep learning or being suitable for all types of noises. It also comes with the overall challenge of developing a good model that can handle such large variation in Bangla texts and variety.

Building upon those studies, Chowdhury et al., (ibid)[6] broaden the usage of sophisticated NLP models for Bangla (2024) evaluating the performance of different models in detecting depression-related content from Bengali social media text. This paper studies similarly powerful models, large language models (such as GPT-3). 5, GPT-4 and a fine-tuned model DepGPT along with deep learning models like LSTM, Bi-LSTM, GRU,Bi-GRU as well as transformer based BERT. BanglaBERT (dataset Ghosh et al.), SahajBERT [3] To this end, the data set — Bengali Social Media Depressive Data Set (BSMDD) is translated and labeled accurately to establish an ideal bench mark for evaluating these models. The implications of this study are, SahajBERT and Bi-LSTM with FastText embeddings showing a prominent outcome on their domain. Perceptually, DepGPT does showcase drastic improvements over GPT-3 — its fine-tuned counterpart. Among all, model 5 enjoyed top position in terms of accuracy (0.9796) and F1-score(0.9804), proving its proficiency exploited with zero-shot few-shot situations prior to any optimization.. Limitations of strengths include potential biases in the social media data and difficulty detecting nuanced clinical expressions of depression. This is a significant improvement over earlier transformer models, while traditional transformable bringing competitive results and that it demonstrates the adaptability of

LLMs in general—DepGPT specifically—a decent endeavour. The novel methods in this research provide critical guidelines to leverage these models for accurate mental health diagnostics, and highlight the specific area of early depression detection through social media. Future research could further refine these models and explore their applicability across different languages and cultural contexts, addressing ethical considerations and data biases to enhance the robustness and reliability of depression detection systems.

Now turning toward the use of machine learning methods to solve the problem of sentiment classification of Bangla texts, Bhowmik et al.[7] proposed an efficient system aimed at counting the sentiment of a sentence by means of a supervised machine learning system and an additional lexicon dictionary. For this purpose, the authors of the study created a lexicon data dictionary (LDD) for the particular area of interest and further proposed a new algorithm, the Bangla Text Sentiment Score (BTSC), to extract meaningful sentiment out of the text. The process involved normalizing, tokenizing, and stemming of the two selected datasets: cricket and restaurant reviews. The BTSC algorithm also helps to apply parts of speech tagging and special characters to be included to aid in calculating sentiment scores. The term frequency-inverse document frequency (tf-idf) was then calculated for the researchers and supervised machine learning classifiers like Support Vector Machine (SVM) were applied. Counting Two-Gram Features proved the efficiency of the BTSC algorithm as SVM achieved 82.21% for bi-gram features. The novelty of the research is in the way it adds an empirical dimension to the usual rule based approach to sentiment analysis and constructs an extensive Bangla sentiment lexicon which is an underdeveloped area in Bangla NLP. The strength of the BTSC algorithm is that it incorporates several linguistic elements, making it more effective in sentiment detection. On the downside, the research depended more on dictionaries created manually and the datasets used were too narrow. Analysis of the results revealed a lower-degree accuracy in relation to the neutral sentiments suggesting that the algorithm and the dictionaries need additional improvements. This research would be of interest because it concentrates on Sentiment Analysis (SA) of the Bangla language which has been ignored in several studies. Similar studies have made use of lexicon-based processes and machine learning for other languages, but the use to Bangla with a thorough feature extraction process is critical. The outcomes of the research conform to the literature which stresses the necessity of having appropriate language resources specific for SA. This research is important due to its relevance to Bangla NLP and also possible usage in social media analysis, opinion mining, and customer feedback systems. For instance, further research may focus on developing the lexicon dictionary, adding new and diverse datasets, and optimizing the BTSC model for neutral sentiment classification. In addition, the application of deep learning methods will also improve the effectiveness and flexibility of the model.

Another paper by Mark Hoogendoorn [8] has conducted research on therapeutic email conversations in order to predict treatment outcomes for social anxiety disorder patients by observing word usages, topics discussed, sentiments and writing style with the help of machine learning algorithms that generates predictive models with promising outcomes.This study is conducted on a dataset of 69 patients who

have mental disorders/anxiety, and it is stated that halfway through the therapy, outcome can be predicted with an Area Under the Curve (AUC) of 0.83, and 0.78 while using the full dataset. The paper also mentions previous research conducted on this topic using Natural Language Processing (NLP) to gain useful information from the writings of the patients. The mentioned 69 patients engaged in a controlled trial which was randomized on a self-help program (internet-guided) which was based on the cognitive-behavioral model by Clark and Wells and the participants received support/feedback through emails from the therapists weekly. The dataset contains the socio-demographic data of the patients and was collected from participants from Switzerland, Austria, and Germany via a study website. Social Phobia Measure was the outcome measure and patients were sectioned into 2 parts-those showing reliable improvement and those not showing. Various algorithms are used in this research to analyze different aspects of the email such as basic mailing behavior (response rates,the length of received emails and response times), word usage (tokenizing words, stemming words, removing stop words, counting occurance), writing style (e.g., part of speech tags), sentiment analysis, topic modeling using Latent Dirichlet Allocation (LDA) and aggregation. The authors compared 'socio demographic data versus email data' for 12 weeks using different algorithms and the results showed promising outcomes in predicting therapeutic outcomes from email conversations. Though they have admitted that the limitation of this research is the fewer number of patients were involved in the research.From this paper, authors give us valuable informations to use in our own research such as: How to perform sentiment analysis, how we can gain clean data from email conversations by using different techniques, how these insights will help us to detect mental health condition.

Muskan Garg's[9] paper has proposed a detailed survey on quantifying(measuring) mental health on social media using machine learning and NLP models. It contains a collection of previous works on suicide detection through a repository. Basically the paper includes recent advancements in AI models, feature extraction and classification, publicly available datasets, and future research directions, challenges faced, hyperbolic geometry for mental health analysis in order to discuss how to handle social media data for stress detection, depression analysis and suicide identification. The author has performed in depth research for 92 research articles (9 articles about stress, 32 were about depression, 37 for suicide risks and 14 of them were for mental disorders). Moreover, the author states that the research has collected necessary features from social media (Twitter, Reddit, Instagram, Facebook) and performed necessary tasks to handle ambiguity of the features. These features can be parted into 4 sections- user profile features (demographic information, user network information), linguistic features (nouns, verbs, and adverbs), social features (post length, hashtags, URLs), and multimedia features. Moreover, sentiment and emotion models like EmoBERT and MentalBERT; lexical features (TFIDF, n-grams), dictionary-based features (LIWC, Suicide dictionary), and syntactic features (Part-Of-Speech tagging); topic modeling methods like LDA; Valence Arousal Dominance (VAD) and Plutchik model - these models were widely used in this research. In addition, for feature vectorization, TFIDF vectorizer, Count vectorizer, Hashing vectorizer, and word-to-vector conversion methods like word2vec, GloVe, and Fasttext; Dimensionality reduction techniques like PCA, NMF, and UMAP were used

for better feature representation. Furthermore, several available datasets were used in this survey such as CLPsych, Reddit Self-reported Depression Diagnosis, MDDL dataset, SMHD dataset, eRISK dataset and others. The paper also provides a overview of automated learning tools used for quantifying mental health in social media datasets and recent developments that are achieved in classification models to identify suicidal tendencies and other mental disorders. This study is very useful for our own research topic as we get to know precious information about sources of databases, linguistic, social, lexical, syntactic features to use, about feature vectorization methods, recent developments related to our research topic - these valuable details have definitely helped us to enrich more knowledge about our topic.

Fernando Arias and Mayteé Zambrano Núñez[10] authored a paper that mentions machine's capability of matching human linguistics with the help of Natural language processing (NLP) from social media data. The paper provides detailed information of various significant concepts of sentiment analysis, extraction and classification of social media data to check mental health degradation over the duration of COVID-19 pandemic. During the COVID-19 pandemic, people tended to spend more time on social media, so Sentiment Analysis technology had a great part in detecting mental health conditions from social media posts. This document shares an overview of how SA technology played a vital role on mental health during the worldwide pandemic days. This study focuses on the enhancement of classification methodologies by separating personal user data from institutional data which helps to study an individual user's mental health condition. For preprocessing, text filtering, noise removal, normalization, and tokenization, Geolocation coding and language selection methods have been used. stemming and TF-IDF models have helped in the case of complex preprocessing, whereas LIWC, LDA, LSA, NMF, word2vec, GloVe, n-gram, PCA, t-SNE, KF, and machine learning techniques- these models have been used for sentiment analysis in this study. Again, Lexicon based methods have efficiently reduced hardiness in the presence of sarcasm language. The authors state that data for sentiment analysis can be collected from Google Trends and can easily be filtered based on the needed criteria, then aspect extraction techniques can be applied to make the data adaptable for subsequent algorithms. Moreover, they mention that opinion classification is performed in order to use available classification algorithms. Prime source of data in this study is text-centered social media like twitter , Facebook and Reddit. Through MonkeyLearn API, Oracle IBM SPSS Modeler, SAS Enterprise Miner and data mining - these social networks can also provide necessary data to analyze.The authors have preferred supervised, semi-supervised, and unsupervised methods in machine learning techniques. Supervised methods such as Naive Bayes, maximum entropy and unsupervised methods like support vector machines and decision trees ;statistical machine learning models, n-gram exploratory analysis techniques, Logistic regression models for evaluating sentiment towards specific factors, SVM, KNN model - these mentioned models were the key models in this study.SA techniques have given accuracy rates up to 69% to detect mental health conditions on social media platforms according to the authors. Similar to the previously discussed four papers, this study also helps us to gain a better understanding about sentiment analysis, preprocessing, how to handle sarcastic language, various NLP models that we can use in our research, sources of data to collect and how we can use these techniques as a whole to perform the whole process.

The paper by Ankit Murarka, Balaji Radhakrishnan, Sushma Ravichandran [11] claims to be the first multi-class model that is based on a transformer-based architect. The paper proposes using a transformer-based language model, RoBERTa (Robustly Optimised BERT Approach) for detecting mental disorders from social media posts. It aimed at detecting and classifying 5 types of mental illnesses, such as, anxiety, depression, bipolar disorder, PTSD (Post Traumatic Stress Disorder) and ADHD (Attention Deficit Hyperactivity Disorder). The authors used the social media platform Reddit for collecting datasets. 13 subreddits and 17159 posts, texts and title texts have been crawled in order to collect these data. 5 out of the 13 subreddits are directly related to the mental illnesses mentioned before. The other 8 subreddits were selected from a variety of topics, that is, music, travel, India, politics, english, datasets, mathematics and science. To boost the performance of the model, the data is augmented using Easy Data Augmentation. In the paper, the performance of the RoBERTa model has been compared to 2 other models, that is, LSTM (Long Short-Term Memory) and BERT, to show the effectiveness of RoBERTa in this case. A recurrent neural network (RNN) that can learn long-term dependencies is termed as LSTM. This model can be used for text recognition, sentiment analysis, language modeling and so on. The authors used NLTK(Natural Language Toolkit) to tokenize the sentences and converted them into lower case to make a vocabulary. Padding and unknown have also been used to describe padding and unknown tokens. Furthermore, words used more than once have been included in the vocabulary. A dropout layer of 0.5 probability was for regularization and the model was trained for 25 epochs. On the other hand, BERT is also a transformer-based language model. A pre-trained tokenizer was used for this model to tokenize the input sentences. A dropout layer with probability of 0.3 was used for regularization for this model . This model was trained for 10 epochs. RoBERTa is an optimized version of the BERT model. This model is better at long range dependencies in texts which can help in text classification where the sentences are interrelated. For this model too, for regularization a dropout layer with probability of 0.3 was used. This model was trained for 10 epochs. The author took the Checklist approach to analyze the model's performance. The authors modified the inputs by replacing words with synonyms and masking tokens to note the change in performance accordingly. They also conducted a Directional Expectation test to check the dependency of the model on certain words. Comparing RoBERTa to the other two models, the paper gives an evident result which shows that the model is more effective than the other models. According to the paper, the model was able to achieve an accuracy rate of 86% for posts and 89% for both posts and titles for detecting mental illness related texts. It also reports the paper had an accuracy rate of 97.5% in detecting non-mental illness posts. Although the accuracy rate is more than 85%, the paper also has some limitations. Firstly, the results obtained in the paper were for the datasets used in the report. So a different dataset could give a different result. Also the datasets were not annotated thus leading to errors in labeling. Secondly, the datasets contained more non-mental illness posts compared to mental illness posts thus making it imbalanced. Thirdly, the author mentioned that the model struggled to detect posts related to depression and anxiety due to vague use of language while describing the two illnesses.This study gives us an overview for multi class detection of mental disorders from social media posts, how to perform this process using the

mentioned BERT model - we can learn how to maintain a high accuracy in detecting mental health condition from texts for our research.

The key finding of another study [12] is that a hybrid classifier using deep learning can accurately identify depression using effective word embedding. The authors claimed to have worked on a new solution by building a new hybrid deep learning neural network named 'Fasttext Convolution Neural Network with Long Short-Term Memory' or in short 'FCL'. The authors worked on 2 sets of data that were collected from Reddit and Twitter respectively. The sentences collected from the posts of the platforms are preprocessed using NLP. The data is preprocessed for tokenization, stop words removal, lower casing, stemming, lemmatization. These methods help in the removal of irrelevant information and the formatting of text data for analysis. To represent the processed data in a more meaningful way, word embedding techniques, such as Word2vec, Glove, fasttext, have been used. CNN and LSTM are two deep learning models that are combined to create a hybrid classifier for depression detection as the paper proposes. The LSTM model can comprehend traits with long-term dependencies, that is, it can identify the sequence of words in a text. It might be able to learn if a sentence is depressive or non-depressive. Again, the CNN model is able to extract global features from texts. So it is able to learn if certain words in a post can be identified as depressive words or not. These two models working together can provide a more accurate classifier that can quickly extract features with dependencies. So this hybrid deep learning classifier makes the detection more accurate. The use of Fasttext with CNN and LSTM or FCL model gives an accuracy rate of 87% for the datasets from Reddit posts, whereas, the accuracy rate for Twitter datasets is 88%. It is to be noted that the accuracy rate was measured upon only 2 social media platforms and English language only. Furthermore the method is limited to detecting depression only in younger groups.As for our research, we will mostly collect data from social medias such as Reddit, Twitter, Facebook and so on. This paper shows us the ways of identifying mental health situation by using word embedding techniques and also we can learn how this research has maintained a higher accuracy throughout the whole process and we intend to implement the same for our own topic.

A paper authored by Danielle Mowery, Craig Bryan and Mike Conway[13] focuses on detecting mental illness from textual data like social media posts taken from Twitter by using natural language processing. The authors used an existing annotated Twitter dataset which was generated based on a hierarchical model of depression related symptoms. The dataset bears 9,473 annotations for over 9,000 tweets. Each tweet is annotated as "no evidence of depression" or "evidence of depression". For tweets annotated "evidence of depression", it is additionally annotated with more dispiriting symptoms. For each class, every annotation is binarized as 1(depressed mood) or 0(not depressed mood). The dataset was encoded with 7 different feature groups such as syntactic features, emotion features,lexical features, demographic features, personality traits,sentiment features and LIWC features. With this feature group, support vector machines performed the best with the highest F1 scores compared to other supervised approaches. The authors eliminated the features that occurred only once in the dataset, which resulted in 5,761 features. They applied Chi-Square feature selection and evaluated their predictive contribution using Sup-

port Vector Machine(SVM) with linear kernel and stratified, 5-fold cross-validation. After the elimination of such features, the authors found that F1 score improved for "depressed mood") from 13 at 1st percentile to 33 at the 20th percentile. They concluded their study experiments by stating that models utilizing reduced feature sets and simple lexical features can produce comparable and in some cases, better output than the larger feature datasets. For our research, we can implement similar datasets with simple lexical features in hope of attaining higher F1 score.

Skaik and Inkpen delve into various NLP techniques for mental health monitoring on social media[14]. They discuss topic modeling, sentiment analysis, and named entity recognition for identifying signs of depression and suicide by analyzing language patterns. The paper also covers machine learning models like CNNs and RNNs used in NLP tasks for mental health surveillance. Their comprehensive search methodology utilizing databases and snowballing yielded relevant publications. Skaik and Inkpen highlight several benefits of social media data: real-time insights for early intervention, large-scale population analysis, and understanding individual experiences. They mention established questionnaires like PHQ-9 and CES-D for depression and SAI/ASI for anxiety, which researchers leveraged to develop NLP models with promising accuracy. For example, Razak et al.'s Tweep, utilizing VADER and CNNs, achieved 68% accuracy in detecting depression. The paper also acknowledges the value of deep learning for automatic feature extraction and emphasizes the ethical considerations of using personal data, advocating for IRB approval. Inspired by this thorough review, we plan to implement these tested methods in our research.

A paper published by Conway et al. discusses using NLP for online mental health monitoring, reviewing studies utilizing diverse sources like Facebook, Twitter, and forums [15]. Similar to other research [14], they highlight the use of SVM, CNN, RNN, and LIWC for mental health detection. The authors emphasize various language characterization metrics like lexical diversity, sentence complexity, and uncertainty for identifying mental health conditions. Notably, one study applied textual cluster analysis to group depression, anxiety, and PTSD. They also commend the shift towards data-driven keyword identification using word embedding for constructing research-appropriate data samples. Inspired by this, we plan to employ these metrics and textual cluster analysis for data categorization in our research.

In another paper Guntuku et al. analyze the potential of social media for mental illness detection, focusing on Twitter data and existing research[16]. They highlight studies using CES-D and BDI scores to assess emotion levels in tweets for depression detection. For example, Tsugawa et al. successfully predicted depression based on CES-D scores and recent tweets. Similarly, the 2015 CLPsych workshop employed topic models and N-grams to identify depression-related words and achieved high prediction performance. Other studies explored psychological dictionaries like LIWC for differentiating mental illness conditions. Notably, some research found machine learning techniques using AUC and PHQ scores achieved results comparable to clinician assessment. However, the authors acknowledge the challenge of depressed individuals potentially reducing social media activity, limiting data streams. They suggest utilizing CES-D and BDI with recent tweet datasets for similar research approaches

Coppersmith et al. explore using NLP and machine learning to detect suicide risk in social media posts[17]. Their proposed system automatically estimates user risk and considers ethical/privacy implications. Inspired by prior research, they leverage the vast amount of user expression on social media to identify potentially suicidal individuals and offer assistance. They acknowledge the trade-off between privacy and prevention, emphasizing user consent for this sensitive technology. This work contributes to mental state forecasting research, highlighting the potential of NLP and ML for suicide prevention and influencing future studies in this area. Notably, the paper introduces innovative NLP/ML techniques for quantifiable sentiment analysis, valuable for forecasting user mental state research.

Islam et al. (2021) explore NLP techniques to detect depression intensity in Bangla social media posts[18]. Collaborating with mental health specialists, they developed a labeled dataset and trained various ML and deep learning models. Their findings are promising, demonstrating these models' effectiveness in detecting depression severity in Bangla. Notably, they emphasize the importance of considering linguistic and cultural factors in mental health detection systems. This research advances NLP for mental health and shows the potential for helping resource-limited individuals. Key takeaways include seeking expert input for accurate labeling and utilizing diverse models for improved accuracy. While the focus on Bangla texts limits its generalizability, it offers valuable insights into analyzing non-English languages with NLP and emphasizes the significance of cultural and linguistic factors in sentiment analysis. This work can inspire future research on incorporating these factors to achieve better results in forecasting users' mental states.

Nijhawan et al. explore leveraging social media posts and comments to forecast stress levels [19]. Recognizing the ubiquity of stress and its varied manifestations, they highlight limitations of traditional detection methods. They then review prior research on sentiment analysis and emotion recognition via NLP and ML. Notably, studies using Twitter data linked negative language and emotion expression to depression. Similar findings emerged for anxiety, with higher first-person pronoun usage and negative emotionality correlating with increased risk. Focusing on stress detection, the authors review studies adopting various ML approaches. One such study, using a Support Vector Machine algorithm, achieved 81% accuracy in predicting stress from social media activity. The paper also acknowledges the growing popularity of deep learning for stress detection. Ultimately, Nijhawan et al. conclude that NLP and ML hold significant potential for stress detection, emphasizing the value of combining qualitative and quantitative methods for accurate insights into social and emotional factors impacting mental health.

The paper "Predicting Depression Levels Using Social Media Posts"[20] explores the potential of social media for predicting users' depression levels. Social media's value in understanding mental health is noted, as users share personal experiences online. Traditional diagnosis difficulties due to limited resources and time-consuming methods motivate this exploration. The authors cite studies linking specific linguistic features in users' tweets, like use of first pronouns and negative emotion words, to depression. Their analysis of tweets using linguistic variables and machine learn-

ing algorithms achieved an 80% accuracy in predicting depression levels, showing promise. Potential ethical concerns regarding social media data and user privacy are also addressed, including the risk of under- or overrepresentation of certain groups. Overall, this research provides a compelling case for using social media data to gain insights into mental health status while raising important ethical considerations. The use of SVM and Naive Bayes classifiers further demonstrates the potential of social media activities to reveal early signs of mental illness.

Another paper authored by Chang Su, Zhenxing Xu, Jyotishman Pathak, and Fei Wang[21] discusses the implementation of machine learning in mental health research. Deep learning model architectures such as Deep Feedforward Neural Network (DFNN), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and autoencoders and their utilization in NLP were mentioned. The authors mentioned that RNN is largely used by researchers to extract textual content from social media posts and detect mental illness. Researchers also came up with the idea of hierarchical RNN architecture equipped with an attention mechanism in order to predict types of posts such as depression, anxiety,autism etc. Authors Prasetio et al. used CNN to detect stress levels from facial frontal images. Their CNN model outperformed traditional machine learning models by 7% in case of accuracy of prediction. Similar studies reported to have detected ADHD and ASD by using CNN architecture. Besides these ML architectures, autoencoders are used by researchers. The challenges that researchers faced while conducting the studies were the constraint of labeled data and the limitation of positive samples. It was also mentioned that for collecting data from social media, there is no straightforward way to confirm "true positives" or "true negatives". As for future scopes and research, the authors found that multimodal will be useful, as mental health is a heterogeneous field, researchers will have the chance to detect mental illness from different perspectives such as online behavioral data, physiological signals, and medical imaging. Zou et al. developed a multimodal model which was composed of 2 CNNs for modeling fMRI and sMRI and achieved 69.15% accuracy.

Chapter 3

Dataset Description

For our paper on "Advancing Sentiment Classification in Bangla Text: An Enhanced BERT Approach on the SentNoB Dataset", we have used the SentNoB dataset as our data source. The dataset was created by extracting posts from various social media platforms which were made in Bangla language. It has about 15,000 posts in total, of which more than 12,000 are set aside for training and the final 3,000 are used for validation and testing. As there was imbalance in the data, the imbalance was handled using RandomOverSampler function. We added another 15000 posts by scraping and collecting comments from Youtube and newspaper sites with the primary dataset.

Each post in the dataset has been labeled as 0, 1 and 2 such that 0 is neutral, 1 is positive and 2 is negative. The dataset can be utilized for tasks involving sentiment analysis and mood prediction because of these labels. It is an open access dataset, so it is publicly available. It can be utilized for many natural language processing applications, especially those concerning sentiment classification or sentiment prediction for Bangla texts.

3.1 Data Pre-processing

The primary dataset, SentNoB, was preprocessed and the quality of the data is good. We have preprocessed the manually added dataset initially in order to achieve clean data and get it ready for analysis. To make sure the data is appropriate for additional analysis, preprocessing is a crucial step in NLP. In our case, we eliminated stopwords and punctuation to improve text data quality and lower noise.

Many punctuations were present in the manually added dataset, although they may have added extra noise and did not add to the text's semantic significance. In tasks like sentiment analysis, where the emphasis is on interpreting the sentiment tone or meaning of the text rather than its grammatical accuracy, punctuation marks like ['—', ',', ';', '?'] are unimportant. By removing these punctuations, the model is able to concentrate on the main ideas in the text. The data also contained emojis which were also removed in the pre-processing stage.

Data	Label
মুগ্ধ হয়ে গেলাম মামু. আর তোমায় কি কমু. বলো তোমায় কোথায় পামু. আমি তোমার সাথে য	1
এই কুত্তার বাচ্চাদের জন্য দেশটা আজ এমন অবস্তায় এই তিনটা পুলিশ কে তরে সবার সামনে	2
ভাই আপনার কথাই যাদু রয়েছে	1
উওরটা আমার অনেক ভাল লেগেছে	1
আমার নিজের গাড়ী নিয়ে কি সাজেক যেতে পারবো না ? প্রাইভেট কার নিয়ে ?	0
যেমন : পরীক্ষার রেজাল্টের সময় , বিভিন্ন ব্যানিজ্যিক প্রচার ইত্যাদি	0
বিশ্বনন্দিত বিশ্ব জয় করা ইসলামের পাখিদের কোরআনের পাখিদের কোন খবর নাই আরিফ ত	2
সাপ্তাহিক মুসুল্লিদের কথাটাও তুলে ধরলে ভাই , চোখ খুলে দিছো	1
আমি ভেবেছিলাম গালিগালাজ করে সেরা ছেলে সে ? গালিবয়	0
তুমি রেপারই হও , ডাক্তার হওয়ার দরকার নাই তোমার	0
লেসবো মেয়েটার ফ্রেন্ড টা অনেক কিউট	2
আচ্ছা চারপাশে অন্ধকার কেন ? লাইট টা শুধু আপনাদের টেবিলে পড়ছে কেন	1
আল্লাহ আপনাকে হায়াত দারাজ করুন এবং সবার মাঝে সুস্থ ভাবে বেঁচে থাকার তৌফিক দান	1
ভাই এতো সুন্দর ভিডিও বানান কিন্তু সাবস্ক্রাইবার বাড়ে না কেনো ?	1
সাধারন মানুষ কি অন্যাই কাজ করে । সব করে রাজনীতি ব্যাক্তি আর পুলিশ । আমার এক্তাই প	2
ভাই দয়াকরে খাবার নষ্ট করবেন্না	2
এইসব ফকির ফাকরা নাকি রেপার হইবো কন্ঠ শুনে মনে হয় ডেন্ডি খায় বাল পাকনা পোলা ক্লা	2
এই চ্যানেল এর খবর গুলা আমার ভালো লাগে । কারন তারা যতটুকুম মারে সত্তি খবর ই দেয়	1
এবার হিজরাদের নিয়ে রিপোর্ট করেন ওদের যন্ত্রণায় মানুষ অতিষ্ঠ হয়ে আছে	1

Figure 3.1: Labeled Dataset

Along with punctuations, we eliminated stopwords, that is, common words considered unnecessary for the majority of NLP activities. Despite being used often in texts, these words have little actual meaning and can be eliminated to avoid taking precedence over more crucial ones. Every language has a unique collection of stopwords- Bangla is no exception. In order to simplify the data and improve model training performance, these phrases were eliminated because they provide little to no context or significance in sentiment analysis.

3.2 Data Augmentation

The idea of data augmentation is to produce altered versions of preexisting data that may be used to artificially expand the size and variety of a dataset. This is particularly crucial for machine learning, as broad and varied datasets enhance the model's capacity to generalize to previously unobserved data. By adding variability, it lessens overfitting and improves model robustness—especially in cases when the original data set is small. This is especially helpful in NLP applications. In case of NLP, various techniques can be used to augment data such as synonym replacement, back translation, inserting noise etc. For our paper, we have used back-translation technique.

Back translation is a data augmentation approach that involves translating the source text into another language and then back into the original(source) language. It is particularly useful in natural language processing (NLP) to provide fresh training data. The first step in back translation for Bangla is to choose an original Bangla

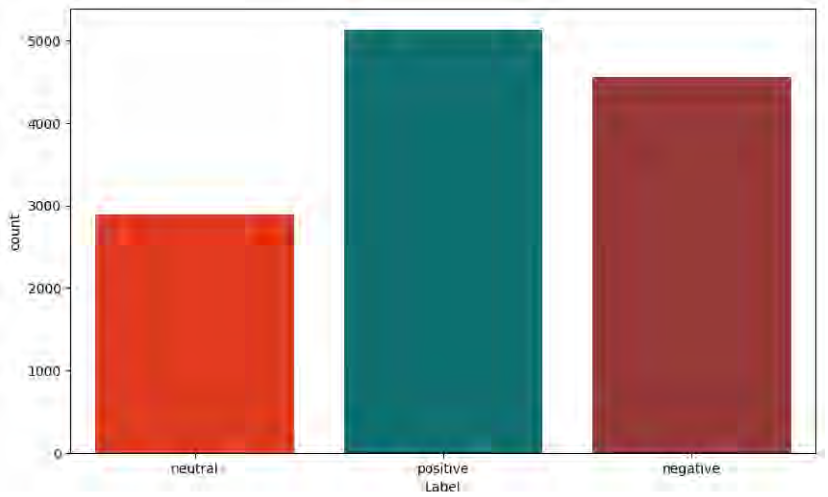

Figure 3.2: Label Distribution of SentNoB Dataset

text to use as the input. Then, a machine translation program is used to convert this content into English. The text is translated, then translated again into Bangla. Though the wording and sentence structure of the final content may change somewhat from the source, the primary idea will still be conveyed. By adding a variety of language variances and producing additional training instances without manual labeling, back translation improves data augmentation. This improves the models' capacity to generalize to new data by enabling them to handle variations in statements of the same meaning.

In this paper, we augmented the 15000 training data of the primary dataset. We used the back translation method to augment the data which then resulted to 30000 training data.

Back translation has some limitations which may affect the data. Our outputs from back translation resembled the source data quite a bit. It produced slightly altered copies rather than variation, which marginally enhanced the model's learning process.

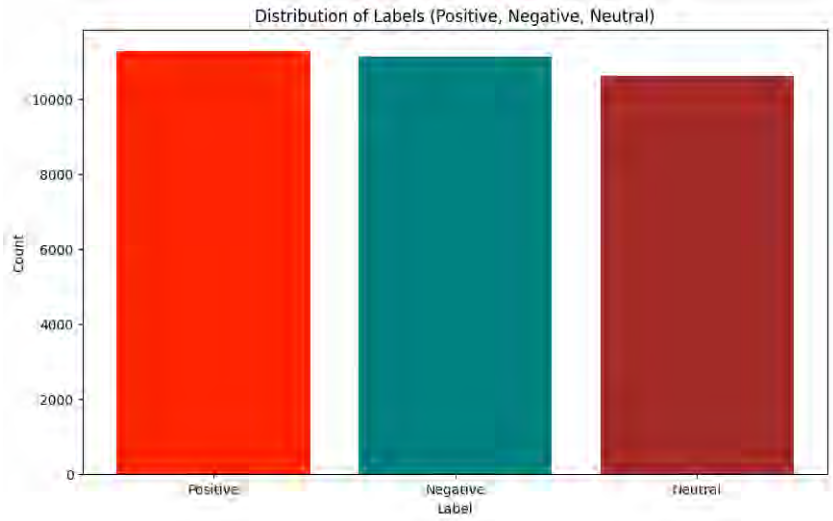

Figure 3.3: Label Distribution of SentNoB and manually added Dataset

Chapter 4

Methodology

4.1 Model Architecture

In this research, we have used BanglaBERT and "Bert-Base-Multilingual-Cased" variant of BERT in order to perform sentiment analysis on our SentNob dataset and augmented dataset. Though BanglaBERT is a BERT based model specifically designed for performing various tasks on Bangla language texts, their model architectures have slight differences.

4.1.1 BanglaBERT

Bangla-BERT is a unique model specifically designed for Natural Language Processing(NLP) tasks in Bangla Language. It is a Transformer-based pre-trained language model and it has been built upon the BERT architecture. This model illustrates the lack of high-quality Bangla-specific-language models. It supports the Transformer architecture - attention mechanisms allow it to capture contextual information more effectively than traditional methods like FastText or Word2Vec. For the pre-training process, it includes a large scale Bangla corpus which enables the model to learn linguistic patterns and semantic structures in the language. By performing fine-tuning, Bangla-BERT gains state-of-the-art performance across various Bangla text classification tasks such as binary sentiment analysis, fake news detection and multiclass sentiment analysis. For our research, we have used Bangla-BERT for its ability to perform sentiment analysis in the case of Bangla texts.
Bangla-BERT leverages the Transformer architecture which contains layers of self-attention and feedforward neural networks. Its core components involve Multi-Head Self-Attention Mechanism which allows the model to focus on different parts of a sentence simultaneously and captures multiple aspects of word relationships. Its Feedforward Neural Network component works after attention is performed- the token representations are passed through fully connected layers in order to perform further processing and feature extraction. Since the Transformer does not have the ability to understand inherent word order, the model adds positional encodings to input tokens to retain sentence structure. Moreover, for Pre-training tasks, Masked Language Modeling(MLM) is used where some words in a sentence are masked and the model predicts them based on the context. In addition, Next Sentence Prediction (NSP) technique is also used- the model predicts whether two sentences follow each other in the corpus. During the stage of fine-tuning, Bangla-BERT is

adapted to specific downstream tasks by adjusting its parameters in order to fit labeled datasets. It performs better than previous models because of its ability to capture deeper linguistic abstractions in Bangla.

4.1.2 BERT

BERT Model has been used here, to be precise the "Bert-Base-Multilingual-Cased" variant of BERT has been used which is a transformer-based model designed to detect the contexts of words in a sentence- it is bidirectional because it checks the preceding and following words in a sentence.

1. **Token Embedding:** BERT splits words into some small subwords using WordPiece tokenization technique where every subword of a text is assigned a numerical value that contains its meaning and context.

2. **Segment Embedding:** BERT generally separates two sentences of a pair by implementing segment embeddings. To explain, BERT takes input as pairs of sentences. Segment embeddings give each sentence a special embedding.

3. **Position Embeddings:** Unlike other sequence models, BERT does not have the ability to know the order of the words in a phrase, but it can understand the sequential nature of words and in order to do that it uses position embeddings to the embedding of every token.

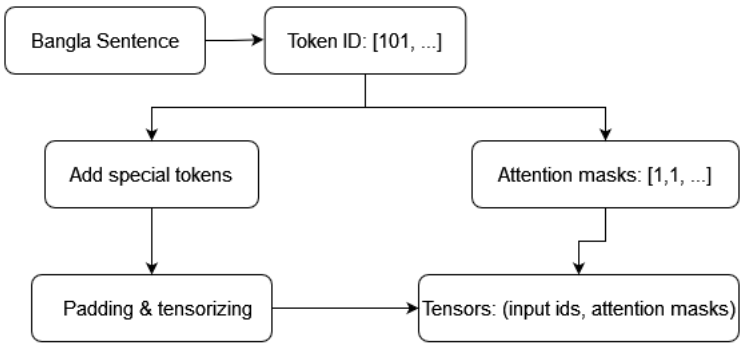

Figure 4.1: Embedding of BERT

- **Transformer Layers of BERT:**

  BERT uses its multi-head self-attention mechanism to calculate the representation of every word and while doing that it focuses on various segments of the given sequence. Several attentions heads are used to help to effectively go through the importance of individual words inside the phrase and thus collects a number of contextual elements.

  BERT then sends the output to a feed-forwarding neural network after the self-attention mechanism processes the sequence of input. This network non-linearly changes the input. For every transformer block, BERT connects layer

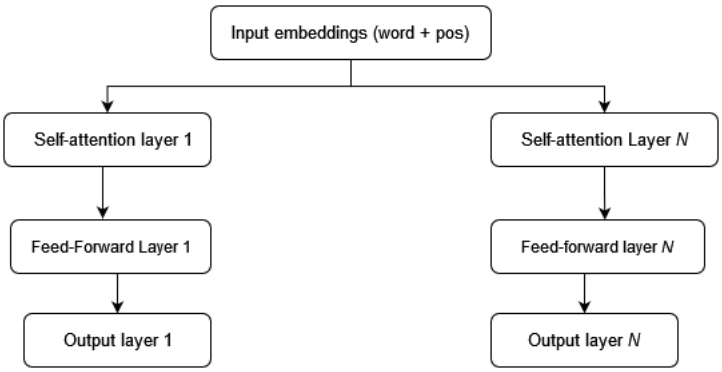

Figure 4.2: Transformer Layers

normalization and residual connections. This step enhances the training step to keep it stable. Through these processes, the gradient problem is taken care of and each layer becomes systematized.

- **Using Tokens for Final Prediction:**

  CLS Token: At the beginning of the input sequence, BERT adds CLS classification token where the final hidden state of this token is considered to be an aggregate representation of the whole input sequence. This representation is the main element in order to make the prediction in text classification or sentiment analysis.

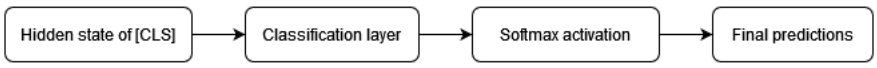

Figure 4.3: Output Representation

  SEP Token: BERT uses SEP token to differentiate between sentences for tasks regarding pairs of sentences which helps the model to understand the structure of the input sequence and how to distinguish between sentences of a pair.

- **Sequence Classification:**

  For sequence classification, first BERT tokenizes input sequences and transforms them into token, position embeddings and segment. After these embeddings are passed through multiple layers of the transformer, BERT identifies semantic and contextual links from the input sequence. The final hidden state which is achieved from the last transformer layer gets passed into a fully connected classification layer- the layer produces the prediction for sequence classification by analyzing the extracted representation into the output classes.

4.1.3 LSTM

The input layer of LSTM receives sequential data like time series, text and data with temporal dependencies. The input shape includes the timesteps and the number of features of timesteps almost every time. The LSTM layers contains memory cells

which can get information of time. The main elements of an LSTM unit are Forget Gate, Input Gate, Cell State and Output Gate. This architecture leads LSTM models to learn long-range dependencies, which is critical for tasks like language modelling, time-series forecasting, and sequential data classification. The final output goes to the dense layers to give a prediction after the data is passed through multiple LSTM layers. For multi class classification problems, a softmax function is applied by the dense layer. In our approach, after adding the embedding layers, we used an LSTM layer with 50 units which learns patterns and dependencies in the input sequence. Then we used a 3 unit output layer that uses the softmax function. As parameters, the initial learning rate was 2e-5, decay steps was 10000, decay rate was 0.96.

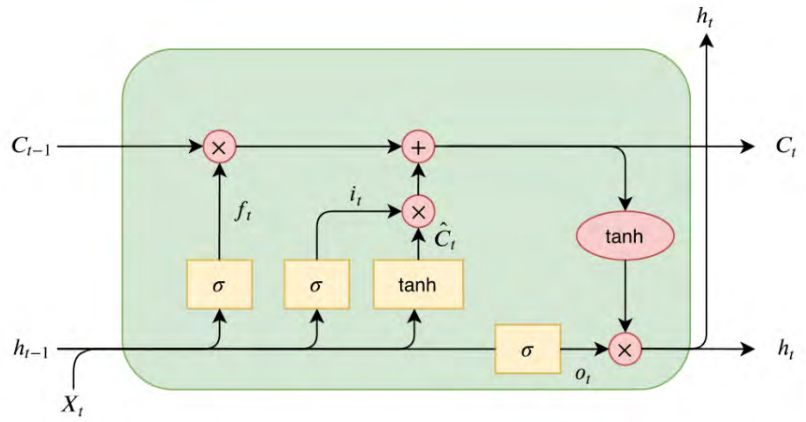

Figure 4.4: LSTM Architecture

4.2 Model Training Initialization

4.2.1 Cross-Validation Setup

To evaluate the model, a 5-Fold Stratified Cross Validation method was used. Using the StratifiedKFold function from sklearn, the data was divided into three folds, each of which preserved the original class distribution. The model was trained on two folds and verified on the third fold for every fold. Every fold was used as the validation set once during the three repetitions of this process.

In order to ensure a comprehensive training procedure, the model underwent 20 epochs of training for every fold. Performance measures, including accuracy, precision, recall, and F1-score, were documented for every fold. After that, the model's final performance was averaged over the three folds to give a trustworthy assessment of its generalisation capability, particularly with regard to handling class imbalances.

4.2.2 Pre-Trained BERT Model Initialization

The paper initializes the 'Bert-base-multilingual-cased' function from Huggingface's 'transformers' library to initialise a pre-trained BERT mode. The 'BertForSequence-Classification' class loads this model, which is intended for multilingual workloads,

in order to do sequence classification. Pre-trained weights speed up training and enhance performance by assisting the model in capturing complex linguistic patterns. Afterwards, the AdamW optimiser and a learning rate scheduler are used to optimise the BERT model on the particular Bengali dataset. For classification tasks, this configuration enables BERT to adapt to the subtleties of Bengali social media posts.

4.2.3 Early Stopping Initialization

During training, an early stopping mechanism is put in place to keep track of the validity loss. The training procedure is terminated early if the validation loss does not improve for a certain number of consecutive epochs. This stops training after performance reaches a limit, thus preventing the model from overfitting.

This paper ensures that training only proceeds if the model's performance on the validation set is getting better. By conserving time and maintaining the optimal model state, this strategy maximises the training process.

4.2.4 Workflow

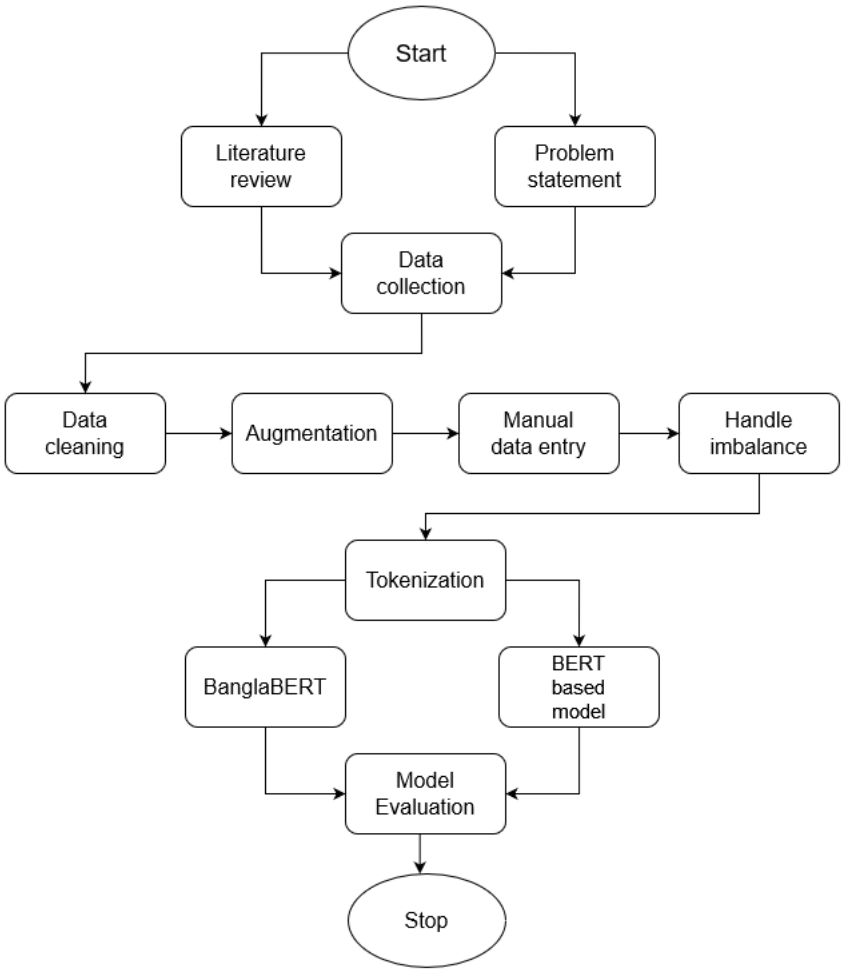

Figure 4.5: Workflow

Chapter 5

Result Analysis

5.1 Training Phase

Classification reports are the illustration of the evaluation of the model's performance that shows insights into various metrics (recall, precision, f1-score) for each of the classes. These metrics help us to achieve valuable information about the model's capability of classifying instances accurately for different categories.

5.1.1 Fold-Wise Classification: BERT

1. **For SentNoB Dataset:** After handling the imbalance of the SentNoB dataset, we have trained the model using BERT based Multilingual Cased variant of BERT model. At different folds, we have tried to illustrate the performance of the model which helps to gain an insight about the effectiveness and dependability regarding the classification of Bangla text data. Let's delve into the classification reports for each of the 5 folds:

    - **Fold 1:** Fold 1 has achieved a moderate accuracy of 64.73% over several domains. The result of precision, recall and F1-score act as an evidence for the model's ability to accurately perform sentiment analysis on our training data.

```
Fold 1 Accuracy: 0.6473161033797217
Fold 1 Classification Report:
              precision    recall  f1-score   support

           0       0.46      0.40      0.43       578
           1       0.74      0.68      0.71      1027
           2       0.65      0.77      0.71       910

    accuracy                           0.65      2515
   macro avg       0.62      0.62      0.61      2515
weighted avg       0.64      0.65      0.64      2515
```

Figure 5.1: Fold 1 for SentNoB dataset using BERT

Neutral(0): With a precision of 0.46 and a recall of 0.40, the model shows a moderate capability of detecting texts with neutral sentiment:

F1-score of 0.40 supports the statement with 578 cases correctly classified as neutral. It means when the model predicts a text to be neutral, it makes the accurate prediction is around 46% of the time and properly recognizes only 40% of them. The precision- recall balance suggests that the model may overlook some cases of neutral related texts of the language.

Positive(1): In the case of positive sentiment detection, the model has a better F1-score of 71%. It has a precision score of 0.74 and recall score is 0.68 and correctly classifies 1027 instances as positive- hence the model can correctly recognize positive sentiment texts 68% of the time and can give accurate prediction around 74% of the time.

Negative(2): Just like the positive sentiment case, the model has a F1-score of 0.71 for the case of negative sentiment detection among the texts. In this case, recall score is 0.65 and precision score is 0.77- which again shows that the model can correctly recognize negative texts 65% of the time and can give correct prediction 77% of the time. 910 instances are successfully identified as negative sentiment texts.

- **Fold 2:** Fold 2 has achieved a better accuracy than before (76.38%) over multiple domains. The result of precision, recall and f1-score act as an evidence for the model's ability to accurately perform sentiment analysis on our training data.

```
Fold 2 Accuracy: 0.763817097415507
Fold 2 Classification Report:
              precision    recall  f1-score   support

           0       0.58      0.66      0.62       579
           1       0.81      0.79      0.80      1027
           2       0.85      0.80      0.83       909

    accuracy                           0.76      2515
   macro avg       0.75      0.75      0.75      2515
weighted avg       0.77      0.76      0.77      2515
```

Figure 5.2: Fold 2 for SentNoB dataset using BERT

Neutral(0): With a precision of 0.58 and a recall of 0.66, the model shows a good capability of recognizing texts with neutral sentiment: F1-score of 0.62 supports the statement with 579 cases correctly detected as neutral sentiment texts. It means when the model predicts a text to be neutral, it makes the accurate prediction is around 58% of the time and properly recognizes 66% of them. The precision- recall balance suggests that the model may overlook some cases of neutral related texts of the language, but it reduces false positives to a good extent.

Positive(1): In the case of positive sentiment detection, the model has the same F1-score of 80%. It has a precision score of 0.81 but recall score

is 0.79 and correctly classifies 1027 instances as positive- hence the model can correctly recognize positive sentiment texts 79% of the time and can give accurate prediction around 81% of the time.

Negative(2): The model has a F1-score of 0.83 for the case of negative sentiment detection among the texts. In this case, recall score is 0.80 and precision score is 0.85- which again shows that the model can correctly recognize negative texts 80% of the time but can give accurate prediction 85% of the time. 909 instances are successfully identified as negative sentiment texts in fold 2.

- **Fold 3:** Fold 3 has achieved a moderate accuracy of 82.10% over several domains.

```
Fold 3 Accuracy: 0.8210735586481114
Fold 3 Classification Report:
              precision    recall  f1-score   support

           0       0.66      0.78      0.71       579
           1       0.89      0.80      0.84      1027
           2       0.88      0.88      0.88       909

    accuracy                           0.82      2515
   macro avg       0.81      0.82      0.81      2515
weighted avg       0.83      0.82      0.82      2515
```

Figure 5.3: Fold 3 for SentNoB dataset using BERT

Neutral(0): With a precision of 0.66 and a recall of 0.78, the model suggests a slightly good capability of recognizing texts with neutral sentiment: F1-score of 0.71 supports the statement with 579 cases correctly detected as neutral sentiment texts-. It means when the model predicts a text to be neutral, it makes the accurate prediction is around 66% of the time. The model properly recognizes 78% of them among the texts.

Positive(1): In the case of positive sentiment detection, the model has a good F1-score of 84%. It has a precision score of 0.89 but recall score is 0.80 and correctly classifies 1027 instances as positive- hence the model can correctly detect positive sentiment texts 80% of the time and can give accurate prediction around 89% of the time.

Negative(2): The model has a F1-score of 0.88 for the case of negative sentiment detection among the texts, which is a strong F1-score. In this case, both recall score and precision score is 0.88 -which means that the model can correctly recognize and correctly give prediction of negative texts 88% of the time. 909 instances are successfully identified as negative sentiment texts in fold 3.

- **Fold 4:** Fold-4 has been able to gain a comparatively better accuracy of 87.35% - precision score, recall and F1-score act as evidence for this

accuracy.

```
Fold 4 Accuracy: 0.873558648111332
Fold 4 Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.76      0.76       579
           1       0.91      0.89      0.90      1026
           2       0.91      0.93      0.92       910

    accuracy                           0.87      2515
   macro avg       0.86      0.86      0.86      2515
weighted avg       0.87      0.87      0.87      2515
```

Figure 5.4: Fold 4 for SentNoB dataset using BERT

Neutral(0): Model has achieved precision score of 0.76 as well as recall of 0.76, the model suggests a moderately good capability of recognizing texts with neutral sentiment: F1-score of 0.76 supports the statement with 579 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction is around 76% of the time.

Positive(1): In the case of positive sentiment detection, the model has a good F1-score of 90%. It has a precision score of 0.91 but recall score is 0.89 and correctly classifies 1026 instances as positive- hence the model can correctly detect positive sentiment texts 89% of the time and can give accurate prediction around 91% of the time.

Negative(2): The model has a F1-score of 0.92 for the case of negative sentiment detection among the texts, which is a strong F1-score. In this case, recall score is 0.93 and precision score is 0.91 -which means that the model can correctly recognize negative texts 93% of the time and correctly gives prediction of negative texts 91% of the time. 910 instances are successfully identified as negative sentiment texts in fold 4.

- **Fold 5:** Fold-5 has achieved the best accuracy of 93.95% which shows a strong capability of the model evidenced by good precision and recall scores.

  Neutral(0): Model has achieved precision score of 0.89 as well as recall score of 0.86; the model suggests a very good ability of recognizing texts with neutral sentiment: F1-score of 0.88 supports the statement with 579 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction is around 89% of the time. The model properly recognizes 86% of them among the texts.

  Positive(1): In the case of positive sentiment detection, the model has a good F1-score of 95%. It has a precision score of 0.95 and similarly recall

```
Fold 5 Accuracy: 0.9395626242544731
Fold 5 Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.86      0.88       579
           1       0.95      0.95      0.95      1026
           2       0.95      0.98      0.96       910

    accuracy                           0.94      2515
   macro avg       0.93      0.93      0.93      2515
weighted avg       0.94      0.94      0.94      2515
```

Figure 5.5: Fold 5 for SentNoB dataset using BERT

score is 0.95 and correctly classifies 1026 instances as positive- hence the model can correctly detect positive sentiment texts 95% of the time and can give accurate prediction around 95% of the time.

Negative(2): The model has a F1-score of 0.96 for the case of negative sentiment detection among the texts, which is a very strong F1-score. In this case, recall score is very high (0.95) and precision score is higher (0.98) -which means that the model can correctly recognize negative texts 95% of the time and correctly gives prediction of negative texts 98% of the time. 910 instances are successfully identified as negative sentiment texts in fold 5.

2. **For Manually Added Dataset:** After handling the imbalance of the SentNoB dataset, we have manually added another 15000 data (total 30,000) with it and then we have trained the model using BERT based Multilingual Cased variant of BERT model. At different folds, we have tried to illustrate the performance of the model again. Let's delve into the classification reports for each of the 3 folds:

   - **Fold 1:** Fold 1 has achieved a moderate accuracy of 67.57% over several domains. The result of precision, recall and F1-score act as an evidence for the model's ability to accurately perform sentiment analysis on our training data.

     Neutral(0): With a precision of 0.60 and a recall of 0.75, the model

```
Fold 1 Accuracy: 0.6757788944723618
Fold 1 Classification Report:
              precision    recall  f1-score   support

           0       0.60      0.75      0.66      3298
           1       0.73      0.63      0.68      3334
           2       0.73      0.65      0.69      3318

    accuracy                           0.68      9950
   macro avg       0.69      0.68      0.68      9950
weighted avg       0.69      0.68      0.68      9950
```

Figure 5.6: Fold 1 for Manually dataset using BERT

shows a moderate capability of detecting texts with neutral sentiment:

F1-score of 0.66 supports the statement with 3298 cases correctly classified as neutral- It means when the model predicts a text to be neutral, it makes the accurate prediction around 60% of the time and properly recognizes 75% of them. The precision- recall balance suggests that the model may overlook some cases of neutral related texts of the language.

Positive(1): In the case of positive sentiment detection, the model has a slight better F1-score of 68%. It has a precision score of 0.73 whereas the recall score is 0.63. It correctly classifies 3334 instances as positive- hence the model can correctly recognize positive sentiment texts 63% of the time and can give accurate prediction around 73% of the time.

Negative(2): Just like the positive sentiment case, the model has a slightly better F1-score of 0.69 for the case of negative sentiment detection among the texts: recall score is 0.65 and precision score is 0.73. It shows that the model can correctly recognize negative texts 65% of the time and can give correct prediction 73% of the time. 3318 instances are successfully identified as negative sentiment texts.

- **Fold 2:** Fold-2 has been able to gain a comparatively better accuracy of 77.70% - precision score, recall and F1-score act as evidence for this accuracy.

```
Fold 2 Accuracy: 0.7770854271356784
Fold 2 Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.83      0.76      3298
           1       0.89      0.66      0.76      3333
           2       0.78      0.85      0.81      3319

    accuracy                           0.78      9950
   macro avg       0.79      0.78      0.78      9950
weighted avg       0.79      0.78      0.78      9950
```

Figure 5.7: Fold 2 for Manually dataset using BERT

Neutral(0): Model has achieved precision score of 0.70 and a recall score of 0.83; the model suggests a moderate ability of recognizing texts with neutral sentiment: F1-score of 0.76 supports the statement with 3298 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction around 70% of the time. The model properly recognizes 83% of them among the texts.

Positive(1): In the case of positive sentiment detection, the model the same F1-score od 0.76 like Neutral label. It has a precision score of 0.89 and recall score is 0.66 and correctly classifies 3333 instances as positive- hence the model can correctly detect positive sentiment texts 66% of the time but can give accurate prediction around 89% of the time.

Negative(2): The model has a F1-score of 0.81 for the case of negative sentiment detection among the texts which is a pretty good F1-score. In this case, recall score is 0.85 and precision score is 0.78 -which means that the model can correctly recognize negative texts 85% of the time but correctly gives prediction of negative texts 78% of the time. 3319 instances support this statement.

- **Fold 3:** Fold-3 has been able to gain a comparatively better accuracy of 88.34%- precision score, recall and F1-score act as evidence for this accuracy.

```
Fold 3 Accuracy: 0.8834053673736054
Fold 3 Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.86      0.86      3298
           1       0.89      0.87      0.88      3333
           2       0.90      0.91      0.91      3318

    accuracy                           0.88      9949
   macro avg       0.88      0.88      0.88      9949
weighted avg       0.88      0.88      0.88      9949
```

Figure 5.8: Fold 3 for Manually dataset using BERT

Neutral(0): Model has achieved precision score of 0.87, both the recall score and F1-score is 0.86. The model suggests a good ability of recognizing texts with neutral sentiment: F1-score of 0.86 supports the statement with 3298 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction around 87% of the time. The model properly recognizes 86% of them among the texts.

Positive(1): In the case of positive sentiment detection, the model has an F1-score of 0.88. It has a precision score of 0.89 and recall score is 0.87 and correctly classifies 3333 instances as positive- hence the model can correctly detect positive sentiment texts 87% of the time but can give accurate prediction around 89% of the time.

Negative(2): The model has a F1-score and recall score of 0.91 for the case of negative sentiment detection among the texts which shows strong capability of the model. In this case, precision score is 0.90. So, the model can correctly recognize negative texts 91% of the time but correctly gives prediction of negative texts 90% of the time. 3318 instances support this statement in fold-3.

3. **For Augmented Dataset(45k):** Following section provides comprehension of BERT model's performance in Bengali text data classification. Let us start with classification reports for each fold:

- **Fold 1:** Overall accuracy for fold-1 is 0.746, which means approximately 74.6% of predictions made by BERT model were correct. The whole metric indicates proportion of correctly classified samples out of total samples in the dataset.

```
Fold 1 Accuracy: 0.7457749246691995
Fold 1 Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.75      0.72      5088
           1       0.79      0.70      0.74      5089
           2       0.76      0.79      0.77      5089

    accuracy                           0.75     15266
   macro avg       0.75      0.75      0.75     15266
weighted avg       0.75      0.75      0.75     15266
```

Figure 5.9: Fold 1 for Augmented dataset using BERT

Neutral(0): Model has achieved precision score of 0.70 and a recall score of 0.75; the model suggests a moderate ability of recognizing texts with neutral sentiment: F1-score of 0.72 supports the statement with 5088 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction around 70% of the time. The model properly recognizes 75% of them among the texts.

Positive(1): In the case of positive sentiment detection, the F1-score is 0.74. It has a precision score of 0.79 and recall score is 0.70 and correctly classifies 5089 instances as positive- hence the model can correctly detect positive sentiment texts 70% of the time but can give accurate prediction around 79% of the time.

Negative(2): The model has a F1-score of 0.77 for the case of negative sentiment detection among the texts. In this case, recall score is 0.79 and precision score is 0.76 -which means that the model can correctly recognize negative texts 79% of the time but correctly gives prediction of negative texts 76% of the time. 5089 instances support this statement.

- **Fold 2:** Overall accuracy for fold-2 is 0.875, which means approximately 87.5% of predictions made by BERT model were correct. The whole metric indicates proportion of correctly classified samples out of total samples in the dataset.

Neutral(0): Model has achieved precision score of 0.89 and a recall score of 0.86; the model suggests a moderate ability of recognizing texts with neutral sentiment: F1-score of 0.87 supports the statement with 5089 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction around 89% of the time. The model properly recognizes 86% of them among the texts.

```
Fold 2 Accuracy: 0.875343901480414
Fold 2 Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.86      0.87      5089
           1       0.87      0.86      0.86      5089
           2       0.87      0.91      0.89      5088

    accuracy                           0.88     15266
   macro avg       0.88      0.88      0.88     15266
weighted avg       0.88      0.88      0.88     15266
```

Figure 5.10: Fold 2 for Augmented dataset using BERT

Positive(1): In the case of positive sentiment detection, the F1-score is 0.86. It has a precision score of 0.87 and recall score is 0.86 and correctly classifies 5089 instances as positive- hence the model can correctly detect positive sentiment texts 87% of the time but can give accurate prediction around 86% of the time.

Negative(2): The model has a F1-score of 0.89 for the case of negative sentiment detection among the texts. In this case, recall score is 0.91 and precision score is 0.87 -which means that the model can correctly recognize negative texts 87% of the time but correctly gives prediction of negative texts 91% of the time. 5088 instances support this statement.

- **Fold 3:** Overall accuracy for fold-3 is 0.944, which means approximately 94.4% of predictions made by BERT model were correct. The whole metric indicates proportion of correctly classified samples out of total samples in the dataset.

```
Fold 3 Accuracy: 0.9443862177387659
Fold 3 Classification Report:
              precision    recall  f1-score   support

           0       0.92      0.96      0.94      5089
           1       0.96      0.92      0.94      5088
           2       0.95      0.95      0.95      5089

    accuracy                           0.94     15266
   macro avg       0.94      0.94      0.94     15266
weighted avg       0.94      0.94      0.94     15266
```

Figure 5.11: Fold 3 for Augmented dataset using BERT

Neutral(0): Model has achieved precision score of 0.92 and a recall score of 0.96; the model suggests a moderate ability of recognizing texts with neutral sentiment: F1-score of 0.94 supports the statement with 5089 cases correctly detected as neutral sentiment texts. So, when the model predicts a text to be neutral, it makes the accurate prediction around 92% of the time. The model properly recognizes 96% of them among the texts.

Positive(1): In the case of positive sentiment detection, the F1-score is 0.94. It has a precision score of 0.96 and recall score is 0.92 and correctly

classifies 5088 instances as positive- hence the model can correctly detect positive sentiment texts 96% of the time but can give accurate prediction around 92% of the time.

Negative(2): The model has a F1-score of 0.95 for the case of negative sentiment detection among the texts. In this case, recall score is 0.95 and precision score is 0.95 -which means that the model can correctly recognize negative texts 95% of the time but correctly gives prediction of negative texts 95% of the time. 5089 instances support this statement.

5.1.2 Classification Reports: Bangla-BERT

1. **For Manual Dataset(30k):**

	precision	recall	f1-score	support
neutral	0.5756	0.4958	0.5327	361
positive	0.7942	0.7966	0.7954	654
negative	0.7706	0.8354	0.8017	571
accuracy			0.7421	1586
macro avg	0.7135	0.7093	0.7099	1586
weighted avg	0.7359	0.7421	0.7379	1586

Figure 5.12: Classification Report for Manual dataset used on BanglaBERT

This classification report gives a detailed overview of the model's precision, recall and F1-score of all three classes. For the neutral class, this model has a precision of 0.5756, recall of 0.4958, and F1-score of 0.5327. In contrast, for positive class, this model shows a strong performance with a precision of 0.7942 and recall of 0.7966 and F1-score of 0.7954. Similarly, in negative class, this model shows excellent performance with precision of 0.7706, recall of 0.8354 and F1-score of 0.8017. If we compare the classes, neutral class has the struggling predictions. It not only misclassifies neutral instances, it also has difficulty in retrieving true neutral cases, which results in the lowest F1-score of all three classes. We can also see, the model is effective at retrieving most of the true negative classes due to its high recall. It also has high F1-score among all, which indicates a good balance between precision and recall in predicting negatives. This model has an overall accuracy of 74.21%, which shows how well the model is performing among all the classes. The macro average precision, recall, and F1-score (which equally weights each class) are 0.7135, 0.7093, and 0.7099, respectively, indicating the model's balanced performance when treating each class equally. Also, the weighted average has precision, recall, and F1-scores of 0.7359, 0.7421, and 0.7379, respectively, showing the model performs slightly better on more frequent classes like positive and negative. So, in summery, the model performs best on the positive and negative classes, achieving high F1-scores, while neutral is the most challenging class for the model, reflected in its lower precision, recall, and F1-score.

2. **For Augmented Dataset(45k):**

	precision	recall	f1-score	support
neutral	0.5692	0.4100	0.4767	361
positive	0.7877	0.8058	0.7967	654
negative	0.7352	0.8459	0.7866	571
accuracy			0.7301	1586
macro avg	0.6974	0.6872	0.6867	1586
weighted avg	0.7191	0.7301	0.7202	1586

Figure 5.13: Classification Report for Augmented dataset used on BanglaBERT

The classification report shows the results of the BanglaBERT model used to a 45k post dataset. Key performance indicators including precision, recall, and F1-score are shown together with an analysis of the model's performance in three categories: neutral, positive, and negative.

Neutral: Out of the three groups, the neutral class performs the worst. Just over half of the neutral predictions made by the model are accurate, with a precision of 56.92%. the recall being 41%, suggesting that the model has a difficult time correctly identifying neutral events and misses a large percentage of them. This under-performance is reflected in the F1-score, which is 47.67% and balances recall and accuracy equally.

Positive: With a precision of 78.77%, the positive class shows significantly better results, indicating that over 79% of the model's positive predictions are accurate. This class has a high recall of 80.58%, meaning that more than 80% of real positive examples are accurately identified by the model. With a 79.67% F1-score for the positive class, accuracy and recall are well-balanced. This shows that the model has a strong capacity to identify positive instances and classify them accurately, indicating that it is extremely good at predicting positive cases.

Negative: The negative class performs admirably although having lesser accuracy than positive class. About 73.5% of the model's negative predictions are true, according to the precision for negative predictions, which is 73.52%. Nonetheless, the recall is greater at 84.59%, suggesting that the model is quite good at identifying real negative cases. The F1-score of 78.66% indicates excellent performance for this class as well as a solid balance between recall and accuracy. While the model correctly detects the majority of negative cases, there is a chance that it may mistakenly classify some examples as negative, as indicated by the larger recall than precision ratio.

With an accuracy of 73.01% overall, the model accurately identifies almost 73% of the test samples. The macro average is 69.74% for accuracy, 68.72% for recall, and 68.67% for F1-score. It is computed by averaging the precision, recall, and F1-score over all three classes without taking into consideration

the class sizes. These numbers point to average performance in each class, with the neutral class performing somewhat differently from the other two. Recall is 73.01%, accuracy is 71.91%, and F1-score is 72.02% after adjusting these values depending on the number of cases in each class using a weighted average. The model performs better for the positive and negative classes, according to these criteria, with the neutral class somewhat depressing the total performance.

5.2 Accuracy Assessment

Analyzing the performance of the models were crucial to understand the effectiveness and dependability of the models. It shows the shortages and capabilities of a model in terms of performing successful sentiment analysis on our Bangla written texts.

5.2.1 Test Accuracy and Evaluation (Using BERT for SentNoB,Manual and Augmented Dataset)

1. **For SentNoB Dataset:** For the case of SentNoB dataset, our test accuracy classification report is shown below:

	precision	recall	f1-score	support
0	0.49	0.48	0.48	361
1	0.78	0.74	0.76	654
2	0.71	0.77	0.74	571
accuracy			0.69	1586
macro avg	0.66	0.66	0.66	1586
weighted avg	0.69	0.69	0.69	1586

Figure 5.14: Testing accuracy of SentNoB using BERT

The core of our accuracy assessment was based on the testing accuracy which reached around 69%. To elaborate, the model correctly classified 69 out of every 100 samples while performing testing. If we look at the class-wise evaluation, we can have a deeper understanding about the overall performance of the model on our test dataset.

Neutral (Label 0): The model achieved a recall and F1 score of 48% for neutral sentiments and also achieved 49% precision score- this means that the model correctly predicted a comment as neutral 49% of the time. Similarly, the model successfully detected 48% of all neutral samples in the test data which suggests it missed more than half of them. Here, low F1 score indicated the model's struggle with neutral comments which is possibly due to overlap with other sentiments. The model correctly identified 361 neutral examples in the test set.

Positive (Label 1): The model performed best in identifying positive sentiments- it has achieved a precision of 78%, a recall of 74% and an F1 score of 76%. So the model made correct predictions 78% of the time and correctly recognized

positive sentiments 74% of the time- it successfully captured 654 positive samples.

Negative (Label 2): For negative sentiments, the model achieved a precision of 71%, a recall of 77% and an F1 score of 74%. This showed that when the model predicted negative comments- 71% of them were indeed negative. The model successfully detected 77% of all negative sentiment texts. The model performed reasonably well in this category but not as strongly as it did with positive sentiments- successfully detecting 571 negative samples.

Hence, the overall accuracy of 69% showed us the general performance of the model but the class based evaluation illustrated how good the model performed in detecting the above mentioned sentiments. The macro average was 66% and weighted average was 69%. The model performs slightly better for the positive and negative sentiments but it needs improvement in detecting neutral sentiment.

2. **For Manual Dataset:** For the case of Manual Dataset, our test accuracy is shown below:

```
Testing Accuracy: 0.6986128625472888
Testing Classification Report:
              precision    recall  f1-score   support

           0       0.50      0.50      0.50       361
           1       0.78      0.73      0.75       654
           2       0.74      0.79      0.76       571

    accuracy                           0.70      1586
   macro avg       0.67      0.67      0.67      1586
weighted avg       0.70      0.70      0.70      1586
```

Figure 5.15: Testing accuracy of Manual dataset using BERT

The model correctly classified 69/70 out of every 100 samples while performing testing.Let's look at the class wise performance:

Neutral (Label 0): The model achieved a recall score, an F1 score and a precision score of 50% for neutral sentiments this means that the model correctly predicted and recognized a comment as neutral 50% of the time- which suggests it missed to detect more than half of them. Here, low F1 score indicated the model's struggle with neutral comments which is possibly due to overlap with other sentiments. The model correctly identified 361 neutral examples in the test set.

Positive (Label 1): The model has achieved a precision of 78%, a recall of 73% and an F1 score of 75%. So the model made correct predictions 78% of the time and correctly recognized positive sentiments 73% of the time and it successfully captured 654 positive samples.

Negative (Label 2): For negative sentiments, the model has performed the best while achieving a precision of 74%, a recall of 79% and an F1 score of

76%. This showed that when the model predicted negative comments- 74% of them were indeed negative. The model successfully detected 79% of all negative sentiment texts. The model performed reasonably well in this category -successfully detecting 571 negative samples.

3. **For Augmented Dataset:**

```
Testing Accuracy: 0.7036569987389659
Testing Classification Report:
              precision    recall  f1-score   support

           0       0.54      0.54      0.54       361
           1       0.77      0.74      0.75       654
           2       0.74      0.77      0.75       571

    accuracy                           0.70      1586
   macro avg       0.68      0.68      0.68      1586
weighted avg       0.70      0.70      0.70      1586
```

Figure 5.16: Testing accuracy of Augmented dataset using BERT

Based on the 45 thousand Bangla sentences, the model achieved accuracy of 70%. From balanced precision and recall, we see that our model effectively classified Bangla text.

With further improvement of Bangla dataset, we believe we can achieve even higher testing accuracy and overall increase in model performance.

5.3 Confusion Matrix Analysis

A model's confusion matrix acts as an important tool to assess the degree to which the model can classify data. This graphical illustration helps to understand model's performance as it allows to see the distribution of accurate and wrong classifications of different classes and gives us a more complete illustration of the overall performance. We can realize the model's decision making capability by analysing the confusion matrix and it unfolds patterns that bring us a picture of the benefits and shortfalls of the model. These informations help us as benchmarks when we try to fine-tune the model and its parameters. So it acts as a parameter and pushes us in the direction of the improvement in terms of the performance.

1. **Confusion Matrix for SentNoB dataset using BERT:**

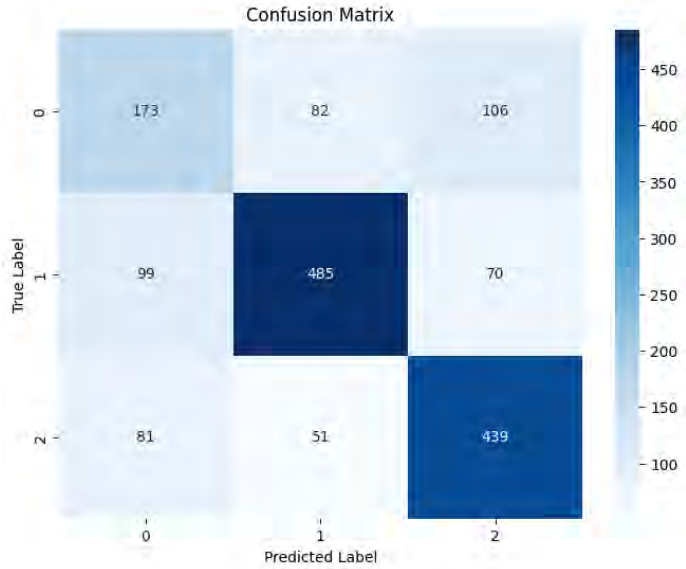

Figure 5.17: Confusion Matrix of SentNoB using BERT

The confusion matrix gives an overview of how well the BERT model was capable of classifying the categories- 0 as neutral, 1 as positive and 2 as negative. For neutral sentiment, the model could correctly predict 173 instances out of 361- it means the model struggled with detecting neutral sentiment evidenced by the misclassification of 82 neutral texts as positive and 106 as negative. For positive label, the model showed its best performance since it could correctly predict 485 positive texts out of 654. Although there were some misplacement of classes with 99 positive comments being misplaced as neutral and 70 as negative, the model overall showed a good performance for positive label. Regarding negative comments, the model correctly classified 439 out of 571, but it misclassified 81 as neutral and 51 as positive. While the model was generally effective at detecting negative comments, it tended to confuse some with neutral ones more than with positive.In the case of negative sentiment, the model correctly identified 439 out of 571, but it confused 81 as neutral and 51 as positive. So the model was pretty much effective at detecting negative comments. This pattern is evidenced by the precision, recall and F1 scores - positive sentiments had the highest accuracy and neutral sentiments struggled to correctly classify.

2. **Confusion matrix for Manual Dataset Using BERT:**

The confusion matrix tells that the model shows strong performance overall- with many cases correctly classified specially for label 1 and label 2. It precisely identified label 1 in 477 instances and label 2 in 452 denoting consistent results for these categories. However, the model did make some notable errors. It incorrectly labeled instances of label 0 as either label 1 or 2 a total of 91 times. There were also cases where label 1 was confused with either label 0 or label 2. Similarly, label 2 was occasionally mistaken for label 0 or 1. All these

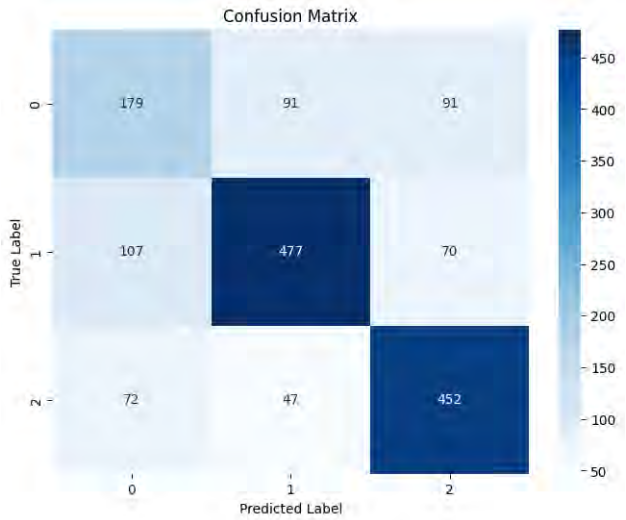

Figure 5.18: Confusion Matrix of Manual dataset using BERT

wrong predictions show us the way where the model could be better especially in differentiating label 0 from the other categories. By brushing up its ability to differentiate between these labels the model could achieve more accuracy and overall reliability.

3. **Confusion Matrix of Augmented Dataset(45k) Using BERT:**

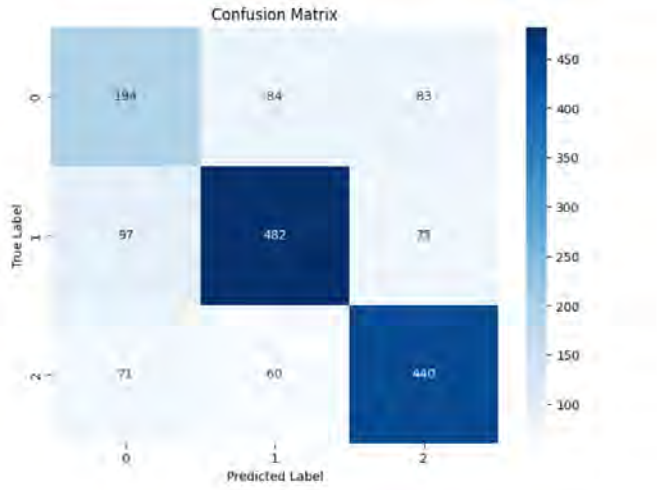

Figure 5.19: Confusion Matrix of Augmented dataset using BERT

This graphical visualization of the model's performance allows us to understand accurate and inaccurate classifications. From the matrix, we can see that most instances are correctly identified which indicates strong overall performance from BERT model. However, there are some falsely identified instances across all the classes. For example, 97 positive sentences were classified as neutral by BERT model. This offers us a chance to further tune our model and dataset to reduce falsely identified classes and increase model accuracy in the future.

4. **Confusion Matrix of Manual Dataset (Using Banglabert)**

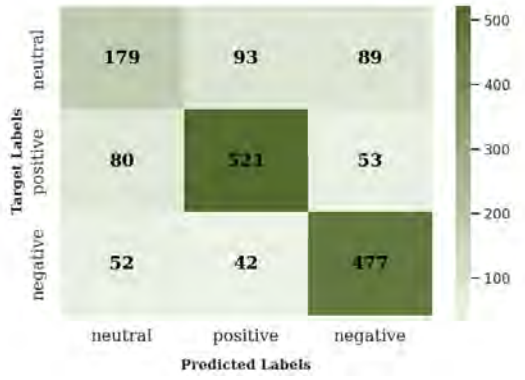

Figure 5.20: Confusion Matrix of Manual dataset using BanglaBERT

For BanglaBERT with a Dataset of 30k, the confusion matrix provides an insightful breakdown among 3 classes, neutral, positive and negative. For neutral class, the model correctly classified 179 samples, and incorrectly classified 93 samples as positive and 89 samples as negative. In case of positive class, the model correctly classified 521 samples and incorrectly classified 80 samples as neutral and 53 samples as negative. Even though, this model classifies positive cases well, it still makes some confusion with other classes, especially neutral class. And finally, for the negative class, 477 samples were correctly classified as negative. However, 52 negative samples were misclassified as neutral and 42 as positive. If we look into the overall scenario, this model faces most challenges while predicting neutral classes where it gets confused with positive and negative classes. This confusion indicates that there are either some class imbalances or there are some similarities between these classes, which makes it harder for the model to accurately differentiate.

5. **Confusion Matrix of Augmented Dataset:**

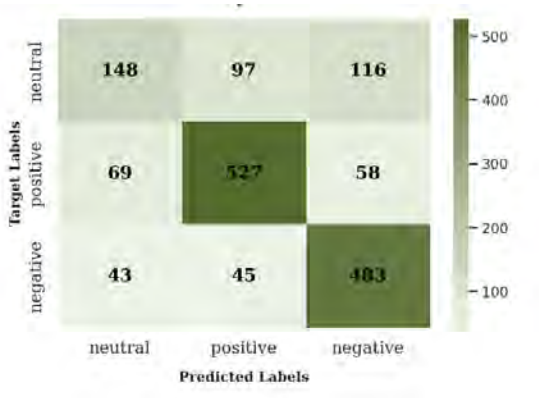

Figure 5.21: Confusion Matrix of Augmented dataset using BanglaBERT

The performance of a sentiment classification model over three categories—neutral, positive, and negative, is represented by the confusion matrix. Every column

indicates the class that the model predicted, and every row represents the actual or target class. The number of accurate predictions for each sentiment is represented by the diagonal values in the matrix. In this case, 483 negative, 527 positive, and 148 neutral instances were all accurately identified by the model. The diagonal values have significant importance as they indicate the model's performance for every class.

The off-diagonal values in the matrix show misclassifications, in which the model incorrectly predicted the sentiment. The model's inability to differentiate neutral state from the others is demonstrated by the 97 cases that were incorrectly labeled as positive and 116 as negative for the neutral class. On the other hand, the positive class performs better and is misclassified less frequently. Merely 69 positive cases were mislabeled as neutral, whereas 58 were anticipated to be negative. This implies that the model has a higher accuracy in detecting sentiment that is positive. In the same way, 45 cases were mistakenly classified as positive and 43 as neutral for the negative class, suggesting that the model does a good job in identifying negative sentiments.
With more accurate predictions and fewer misclassifications, the model generally does the best job of detecting both positive and negative feelings. The neutral class presents additional difficulties, though, as more cases are incorrectly categorized as either positive or negative. This suggests that neutral sentiments may require further refinement, either through additional data or model tuning, as the model may have trouble differentiating between them.

5.4 Training and Validation Losses

1. **Using BERT for SentNoB dataset:**

Figure 5.22: Training loss for SentNoB using BERT

The graph indicates how the training and validation losses change over the whole period of training. Here, the blue line represents the training loss and it starts high (around 0.9) and gradually it is seen to be decreasing by the final epoch- it means the model is learning and improving its accuracy on the

training data. On the other hand, the orange line represents the validation loss which starts at 0.8 and it also decreases over time with more fluctuations reaching 0.3-0.4 around the final epoch. Hence, the model is performing better on training data but the gap between the two lines suggests that the model is slightly underperforming to the validation data.

2. **Using BERT for Manual Dataset (30kdata):**

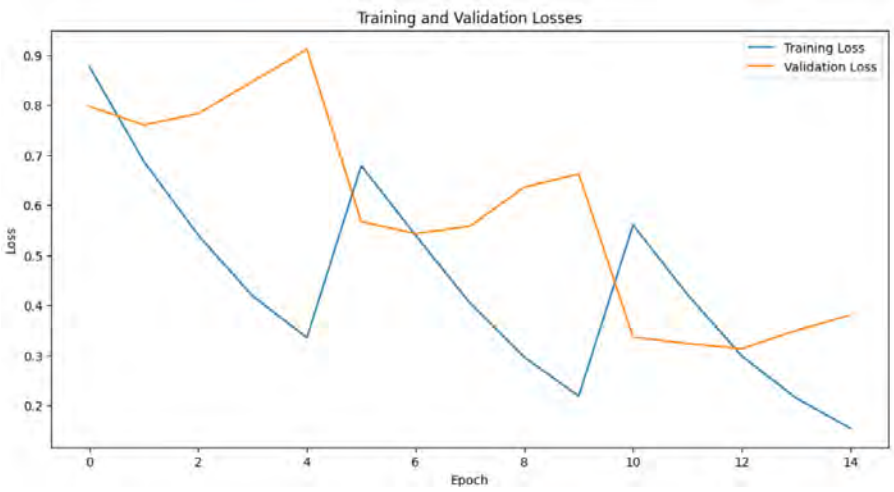

Figure 5.23: Training loss for Manual data using BERT

In the training and validation losses graph the blue line represents the training loss which consistently decreases throughout the epochs even though it started at a higher value of 0.9- indicates that the model is minimizing its errors on the training data. In contrast, the validation loss(the orange line) decreases initially but starts to fluctuate after a few epochs. This pattern which is similar to the validation accuracy suggests that while the model is learning the training data well, it struggles to maintain consistent performance on the validation data.

3. **Using BERT for Augmented Dataset(45k):**

Figure 5.24: Training loss for Augmented data using BERT

Observations from both loss and accuracy plot altogether provides us insight about the model's learning and classification behavior. It also helps us to detect overfitting or underfitting. Ultimately, the goal is to achieve minimum loss and high accuracies for both training and validation, which indicates a generalized model fit for performing effectively on unseen data.

4. **Using BanglaBERT for Manual Dataset:**

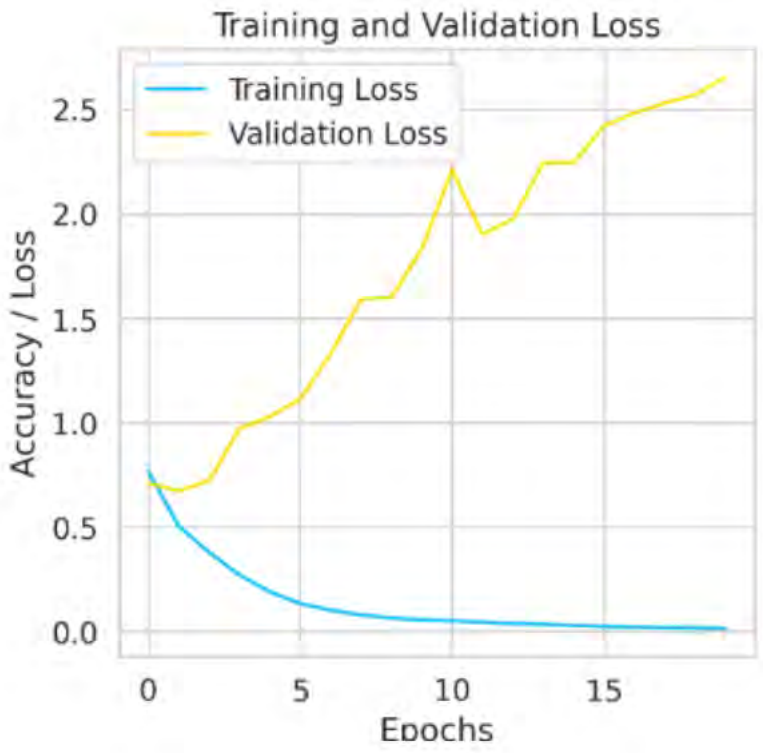

Figure 5.25: Training loss for Manual data using BanglaBERT

Looking at the loss curves, the training loss decreases steadily over time, which is a good sign that the model is learning and fitting the training data well. By around 15 epochs, it seems to converge, meaning the model has adapted to the training set. However, the validation loss tells a different story. It drops initially, but after about 5 epochs, it starts increasing, which suggests that the model might be overfitting—learning too much from the training data and struggling to generalize to new, unseen data.

5. **Using BanglaBERT for Augmented Dataset:**

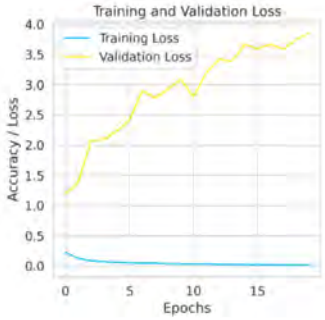

Figure 5.26: Training loss for Augmented data using BanglaBERT

The model's loss behavior over 20 epochs for both the training and validation sets is depicted in the Training and Validation Loss graph. The training loss is shown by the blue line, which gradually drops until it approaches zero, demonstrating how well the model fits the training set. The validation loss, represented by the yellow line, on the other hand, has a steady upward trend, indicating that the model's performance on the validation set deteriorates with more training. Overfitting, in which the model is excessively specialized to the training data but is unable to generalize to fresh, unknown data, is strongly indicated by the widening disparity between the two losses.

5.5 Training and Validation Accuracy

1. **Using BERT for SentNoB Dataset:**

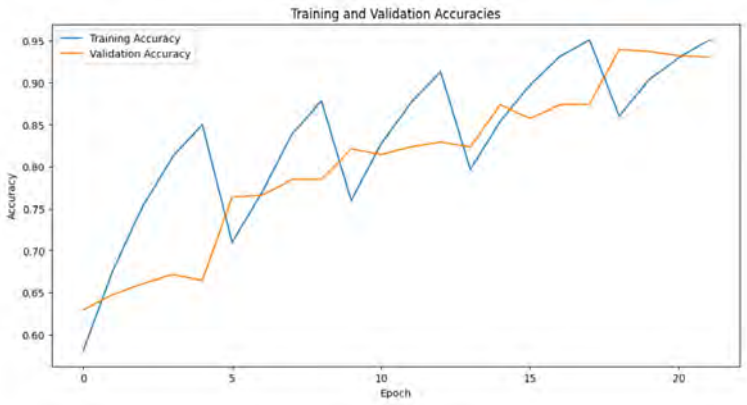

Figure 5.27: Training Accuracy on SentNoB using BERT

The graph has accuracy ranges on the y-axis and epoch numbers on the x-axis. The blue line represents the training accuracy and the orange one represents the validation accuracy. Initially, both the accuracies are relatively low. Gradually, the training accuracy starts to increase in a non-linear fashion and slightly fluctuates at some points and ultimately reaches 95% around the last epoch- showing strong performance on the training data.On the other hand,

the validation accuracy follows a similar pattern but with more fluctuations and it has a lower peak. Around the 10th epoch, it stabilizes but continues to fluctuate and eventually peaks at around 90%. It shows that the model performs well enough on the validation data.

2. **Using BERT for Manual dataset:**

Figure 5.28: Training Accuracy on Manual data using BERT

The blue line represents the training accuracy-it increases steadily as the model learns from the training data and eventually approaches a high value near 0.95. This consistent improvement suggests that the model is effectively learning the patterns in the training data. On the opposite, the validation accuracy(represented by the orange line) initially improves but it starts to fluctuate after a few epochs and eventually levels off around 0.85. So, it performs exceptionally well on the data it was trained on but its performance on validation data does not show the same level of improvement which implies that the model is not generalizing as well as it could.

3. **Using BERT for Augmented Dataset:**

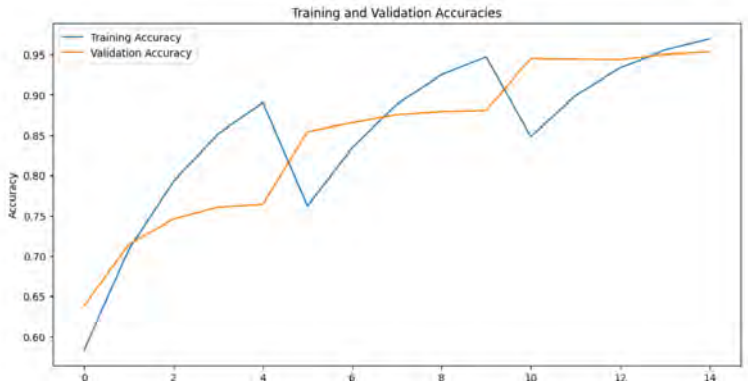

Figure 5.29: Training Accuracy on Augmented data using BERT

Observations from both loss and accuracy plot altogether provides us insight about the model's learning and classification behavior. It also helps us to detect overfitting or underfitting. Ultimately, the goal is to achieve minimum loss and high accuracies for both training and validation, which indicates a generalized model fit for performing effectively on unseen data.

4. **Using BanglaBERT for Manual Dataset:**

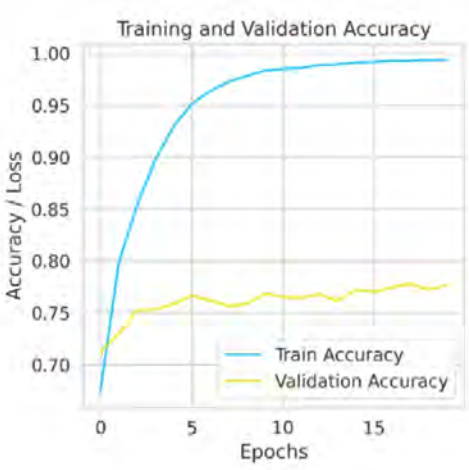

Figure 5.30: Training Accuracy on Manual data using BanglaBERT

In terms of accuracy, the model's training accuracy climbs quickly, reaching nearly 100%, which shows that it performs exceptionally well on the training data. On the other hand, the validation accuracy levels off at around 75%, indicating that the model isn't improving on the validation set. This gap between the training and validation accuracy, combined with the rising validation loss, is another indicator of overfitting, where the model is fitting the training data too closely but isn't generalizing well to new data.

5. **Using BanglaBERT for Augmented Dataset:**

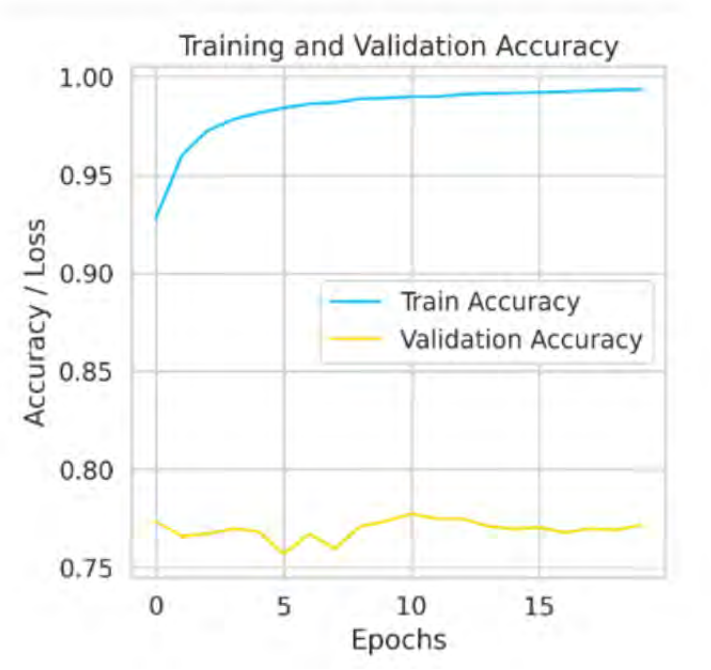

Figure 5.31: Training Accuracy on Augmented data using BanglaBERT

The model's performance across 20 epochs on both the training and validation datasets is displayed in the Training and Validation Accuracy graph.

The training accuracy is represented by the blue line, which rises quickly and reaches a plateau close to 1.00, showing that the model is nearly flawlessly fitting the training set. The validation accuracy, shown by the yellow line, does not increase in the same way over the epochs and instead varies between 0.75 and 0.80. This discrepancy between training and validation accuracy raises the possibility of overfitting since it shows that although the model learns to predict well on the training set, it has trouble generalizing to unknown validation data.

5.6 ROC curve and ROC area for each class

1. **For Manual Dataset Using BanglaBERT:** The ROC curve analysis provides a clear picture of how well the model distinguishes between the three classes. For class 0, the model has an AUC of 0.78, indicating that it's somewhat effective but finds class 0 harder to classify compared to the others. Class 1 fares better with an AUC of 0.89, showing a strong ability to differentiate class 1 instances from others. The best performance is seen with class 2, which has the highest AUC of 0.91, meaning the model excels at predicting this class. Overall, the higher AUC values for classes 1 and 2 reflect better prediction accuracy, while class 0 remains more challenging for the model to handle.

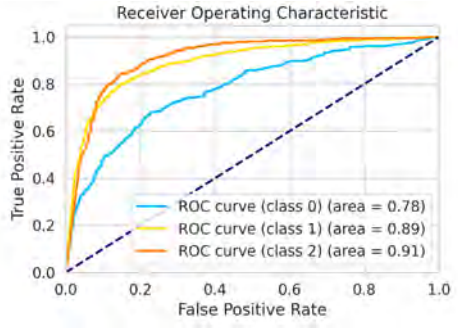

Figure 5.32: ROC curve for Manual Dataset

2. **For Augmented Dataset Using BanglaBERT:** The multi-class classification model's Receiver Operating Characteristic (ROC) curves show three distinct classes—class 0, class 1, and class 2. Each curve shows how well the model performed in differentiating between positive and negative sentiments. On the x-axis, the false positive rate is displayed against the true positive rate on the y-axis. A curve's performance improves with its proximity to the upper-left corner. With an Area Under the Curve (AUC) of 0.90 for class 2, the model performs best; class 1 comes in second with an AUC of 0.89. With an AUC of 0.76, Class 0 performs the worst. The dash diagonal line, with an AUC of 0.5, represents a random classifier. All the other ROC curves being above it, indicates that the model performs better than random guessing for all class.

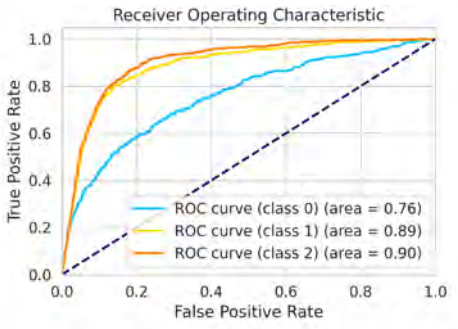

Figure 5.33: ROC curve for Augmented Dataset

5.7 Ensemble Algorithm (LSTM and BanglaBERT) Classification Report

The main motivation behind the ensemble approach was to use the LSTM and BERT together by the system to make effective predictions. The LSTM handles text in sequences to capture the pattern along the temporal dimension. This helps the model in finding some temporal dependencies or sequential relationships present in the input data. Such models are very helpful in tasks where word order and sentence structure both play a very important role.
The BERT model processes this text in a context-sensitive way. It looks at the whole sentence at once and gives a really rich contextualised representation for every word, considering other words that set the scene around it. This helps in capturing deep semantic meaning.
This code takes predictions from two models. Then it combines the predictions. For deciding the final label, it uses majority voting. Lastly evaluates the ensemble's performance.

5.7.1 Ensemble Result:

	precision	recall	f1-score	support
0	0.47	0.63	0.54	361
1	0.79	0.72	0.75	654
2	0.79	0.70	0.74	571
accuracy			0.69	1586
macro avg	0.68	0.68	0.68	1586
weighted avg	0.72	0.69	0.70	1586

Figure 5.34: Ensemble Testing Accuracy

The accuracy of the ensemble model gives an output of 69% which means 69% predictions of all classes are correct. The model shows a variety of effects in different classes. The precision, recall, and F1-scores shows that the model gives more efficient results while classifying examples of class 1 and 2. They have higher precision and recall values. The recall for class 0 is 0.63 which is comparatively better. This means the class 0 instances are correctly identified by the model. While the overall accuracy is high, it means the model is working better on classes 1 and 2. The weighted average is slightly improved than the macro average. The overall ensemble result is lower than the one we got from BanglaBert. It can be said that ensembling LSTM and BanglaBert predictions weakened the overall model performance

5.8 Overall Performance of Models and Comparison:

Let's have an overview of the overall performances of all the models that we have used in this research:

Models	**SentNoB Data**	**SentNoB + Manual Data Entry**	**SentNoB + Augmented + Manual Data Entry**
BanglaBert (Test Accuracy)	0.73	0.7421	0.7301
Bert-base-multilingual-cased (Test Accuracy)	0.69	0.6986	0.70
BanglaBert and LSTM ensembled (Test Accuracy)	0.69		

Table 5.1: Comparison among performance of different datasets on multiple models

After performing back translation to augment data, we did not receive good performance because the source language (Bangla) is being translated to English and then translating it back to source language(Bangla)- these steps include translation models which are not accurate for Bangla since they may introduce errors or alter sentiment-carrying words. Again, Bangla is a low-resource language in terms of NLP tools, so this might be another reason which led to unreliable augmented data in our case. Moreover, there might be issues involving loss of cultural and contextual nuances, noise in the augmented data and sentiment shift issues. So, we decided to add data manually while being careful about the data imbalance. With the added manual data, we received slightly good performance (accuracy score 0.7421) using BanglaBERT- which is a slight improvement than existing papers on SentNob using BanglaBERT model. According to the paper [5], the authors have received an accuracy score of 0.72 using BanglaBERT base model. In another paper [3], authors have achieved an accuracy of 72.89% using BanglaBERT for sentiment classification. Moreover, in the paper [2], the authors have achieved an accuracy score of 0.61 while using the model BanglaBERT and tested on SentNob paper; they have also used SentNob as training dataset and achieved 0.70 accuracy on combined dataset and only 0.55 accuracy on their own manual dataset.

Our ensemble algorithm (LSTM+BanglaBERT) has managed to achieve an accuracy of 0.69 over the SentNob dataset.

5.9 Limitations

Our model faces difficulties while identifying sarcastic texts. The ensemble model gives a less effective f1-score of 0.54 in the neutral class compared to the positive class and negative class. Hence, for the neutral class, the model is not as effective as it is in the cases of positive and negative classes. The models we used mostly performed worse on the neutral text prediction.

Chapter 6

Conclusion

In this project, we investigated how NLP techniques, especially, BERT could be used to predict users' sentiment from Bangla-language social media posts. We were able to provide insights into user sentiment by classifying text into neutral, positive and negative sentiments by utilizing the SentNoB dataset, which has over 15,000 data.

We addressed the constraints of a comparatively small dataset and enhanced the generalizability of the model by employing back translation for data augmentation. According to our findings, pre-trained transformer-based models such as BERT and BanglaBERT are quite successful in processing Bangla texts and identifying the subtleties of sentiment in this language. The model's accuracy in predicting the sentiment of Bangla social media posts lays the groundwork for more extensive uses in sentiment analysis, mental health monitoring and user behavior prediction within the Bangla-speaking population.

Future research might concentrate on growing the dataset, improving models for particular mental states other than sentiment and investigating cross-lingual transfer learning to include posts in several languages. Moreover, improving the accuracy of the ensemble algorithm might be another significant issue to work on.These models' integration with practical applications, like social media monitoring tools or chatbots for sentiment analysis- these steps might be useful for monitoring comments on social media business pages to detect which sentiment is trending for a specific post.

Bibliography

[1] T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, "Natural language processing applied to mental illness detection: A narrative review," *NPJ digital medicine*, vol. 5, no. 1, p. 46, 2022.

[2] M. E. Islam, L. Chowdhury, F. A. Khan, *et al.*, "Sentigold: A large bangla gold standard multi-domain sentiment analysis dataset and its evaluation," *arXiv*, 2023. DOI: 10.48550/ARXIV.2306.06147.

[3] A. Bhattacharjee, T. Hasan, W. Ahmad, *et al.*, "Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla," in *Findings of the Association for Computational Linguistics: NAACL 2022*, Association for Computational Linguistics, 2022. [Online]. Available: https://doi.org/10.18653/v1/2022.findings-naacl.98.

[4] A. Le Glaz, Y. Haralambous, D.-H. Kim-Dufor, *et al.*, "Machine learning and natural language processing in mental health: Systematic review," *Journal of Medical Internet Research*, vol. 23, no. 5, e15708, 2021.

[5] K. Elahi, T. Rahman, S. Shahriar, S. Sarker, M. Shawon, and G. M. Shahariar, "A comparative analysis of noise reduction methods in sentiment analysis on noisy Bangla texts," in *Proceedings of the Ninth Workshop on Noisy and User-generated Text (W-NUT 2024)*, R. van der Goot, J. Bak, M. Müller-Eberstein, W. Xu, A. Ritter, and T. Baldwin, Eds., San Ġiljan, Malta: Association for Computational Linguistics, Mar. 2024, pp. 44–57. [Online]. Available: https://aclanthology.org/2024.wnut-1.5.

[6] A. K. Chowdhury, S. R. Sujon, M. S. S. Shafi, *et al.*, "Harnessing large language models over transformer models for detecting bengali depressive social media text: A comprehensive study," *Natural Language Processing Journal*, vol. 7, p. 100 075, 2024. [Online]. Available: https://doi.org/10.1016/j.nlp.2024.100075.

[7] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal, and M. S. Islam, "Bangla text sentiment analysis using supervised machine learning with extended lexicon dictionary," *Natural Language Processing Research*, vol. 1, no. 3-4, p. 34, 2021. [Online]. Available: https://doi.org/10.2991/nlpr.d.210316.001.

[8] M. Hoogendoorn, T. Berger, A. Schulz, T. Stolz, and P. Szolovits, "Predicting social anxiety treatment outcome based on therapeutic email conversations," *IEEE journal of biomedical and health informatics*, vol. 21, no. 5, pp. 1449–1459, 2016.

[9] M. Garg, "Mental health analysis in social media posts: A survey," *Archives of Computational Methods in Engineering*, vol. 30, no. 3, pp. 1819–1842, 2023.

[10] F. Arias, M. Z. Nunez, A. Guerra-Adames, N. Tejedor-Flores, and M. Vargas-Lombardo, "Sentiment analysis of public social media as a tool for health-related topics," *IEEE Access*, vol. 10, pp. 74 850–74 872, 2022.

[11] A. Murarka, B. Radhakrishnan, and S. Ravichandran, "Classification of mental illnesses on social media using roberta," in *Proceedings of the 12th international workshop on health text mining and information analysis*, 2021, pp. 59–68.

[12] V. Tejaswini, K. S. Babu, and B. Sahoo, "Depression detection from social media text analysis using natural language processing techniques and hybrid deep learning model," *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2022.

[13] D. Mowery, C. Bryan, and M. Conway, "Feature studies to inform the classification of depressive symptoms from twitter data for population health," *arXiv preprint arXiv:1701.08229*, 2017.

[14] R. Skaik and D. Inkpen, "Using social media for mental health surveillance: A review," *ACM Computing Surveys (CSUR)*, vol. 53, no. 6, pp. 1–31, 2020.

[15] M. Conway, M. Hu, and W. W. Chapman, "Recent advances in using natural language processing to address public health research questions using social media and consumergenerated data," *Yearbook of medical informatics*, vol. 28, no. 01, pp. 208–217, 2019.

[16] S. C. Guntuku, D. B. Yaden, M. L. Kern, L. H. Ungar, and J. C. Eichstaedt, "Detecting depression and mental illness on social media: An integrative review," *Current Opinion in Behavioral Sciences*, vol. 18, pp. 43–49, 2017.

[17] G. Coppersmith, R. Leary, P. Crutchley, and A. Fine, "Natural language processing of social media as screening for suicide risk," *Biomedical informatics insights*, vol. 10, p. 1 178 222 618 792 860, 2018.

[18] M. K. Kabir, M. Islam, A. N. B. Kabir, A. Haque, and M. K. Rhaman, "Detection of depression severity using bengali social media posts on mental health: Study using natural language processing techniques," *JMIR Formative Research*, vol. 6, no. 9, e36118, 2022.

[19] T. Nijhawan, G. Attigeri, and T. Ananthakrishna, "Stress detection using natural language processing and machine learning over social interactions," *Journal of Big Data*, vol. 9, no. 1, pp. 1–24, 2022.

[20] M. M. Aldarwish and H. F. Ahmad, "Predicting depression levels using social media posts," in *2017 IEEE 13th international Symposium on Autonomous decentralized system (ISADS)*, IEEE, 2017, pp. 277–280.

[21] C. Su, Z. Xu, J. Pathak, and F. Wang, "Deep learning in mental health outcome research: A scoping review," *Translational Psychiatry*, vol. 10, no. 1, p. 116, 2020.