Recognition of Bangla and English Words in Bangla Texts Using a Modified BERT-base-NER Model

By

Md. Parvez Hossain
012221016

Submitted in partial fulfilment of the requirements
of the degree of Master of Science in Computer Science and Engineering

January 3, 2026

Department of Computer Science and Engineering
United International University

Approval Certificate

This thesis titled **'Recognition of Bangla and English Words in Bangla Texts Using a Modified BERT-base-NER Model'** submitted by **'Md. Parvez Hossain'**, Student ID: **'012221016'**, has been accepted as **Satisfactory** in fulfillment of the requirement for the degree of Master of Science in Computer Science and Engineering on 'January 3, 2026'.

Board of Examiners

1.

_____ **Supervisor**

Prof. Dr. Mohammad Nurul Huda
Professor
Department of CSE, United International University

2.

_____ **Head Examiner**

Dr. Khondaker Abdullah -Al-Mamun
Professor
Department of CSE, United International University

3.

_____ **Examiner-I**

Mr. Nahid Hossain
Assistant Professor
Department of CSE, United International University

4.

_____ **Examiner-II**

Ms. Rubaiya Rahtin Khan
Assistant Professor
Department of CSE, United International University

6.

_____ **Ex-Officio**

Dr. Hasan Sarwar
Professor, Dept. of CSE & Dean, SoSE
United International University

Declaration

This is to certify that the work entitled **'Recognition of Bangla and English Words in Bangla Texts Using a Modified BERT-base-NER Model'** is the outcome of the research carried out by me under the supervision of **'Prof. Dr. Mohammad Nurul Huda, Professor, Department of CSE, United International University'**

Md. Parvez Hossain
MSCSE Program
Student ID: 012221016
Dept. of Computer Science and Engineering
United International University
Dhaka, Bangladesh

In my capacity as supervisor of the candidate's thesis, I certify that the above statements are true to the best of my knowledge.

Prof. Dr. Mohammad Nurul Huda
Professor
Dept. of Computer Science and Engineering
United International University
Dhaka, Bangladesh

Abstract

A combination of Bangla and English words is commonly used, particularly on social media. This tendency greatly hampers the next generation's ability to learn Bangla. This study suggests an approach for identifying words in Bangla texts that are both English and Bangla. This study also translates the identified English terms into standard Bangla words. The Transformer architecture, which uses an attention mechanism to identify the connections between words and their contexts inside a text, is the foundation of bidirectional encoder representations from transformers (BERT). In this study, we use the training input dataset to modify the BERT-base-NER model. For the name entity recognition (NER) task, the proposed BERT-base-NER model in this study achieves state-of-the-art performance. For both the training and testing scenarios, we employ a holdout cross-validation procedure. We used 80% of the entire data for training and 20% for testing. We use the Google Translate API (application programming interface) to translate the identified English words into standard Bangla words. In order to assess the modified BERT-base-NER model, we applied the input dataset to the current machine learning (ML) and deep learning (DL) techniques. Support vector machines (SVM) and Naive Bayes (NB) are two components of the machine learning approach. Conversely, the DL method uses bidirectional LSTM (BiLSTM), long short-term memory (LSTM), and convolutional neural network (CNN). The improved BERT-base-NER model is highly accurate and efficient at identifying Bangla and English words, according to simulation data. With an accuracy of 95%, the proposed BERT-base-NER model achieves the best result among the current methods. For Bangla–English code-mixed text, this study presents a reliable BERT-based word-level language identification system that successfully resolves Banglish ambiguity and allows downstream Bangla language processing applications such as standard Bangla conversion, machine translation, and information extraction.

Acknowledgements

I would like to begin by thanking Almighty Allah for His blessings, which helped me to complete my thesis. Without the help, tolerance, and encouragement of numerous people, the study presented in this dissertation would not have been possible. My thesis supervisor, **Prof. Dr. Muhammad Nurul Huda**, Professor, Department of Computer Science and Engineering (CSE), United International University (UIU), deserves special recognition for his unwavering support and assistance, even with his hectic schedule. His insight led to the original proposal to examine the possibility of re-examining the sensitivities of the entire thesis work. I genuinely thank him for his support, since he has helped me through some very difficult times while I've been writing the dissertation and doing the analysis.

I would also like to thank **Dr. Ohidujjaman**, Assistant Professor, Department of CSE, UIU, for his guidance and support. I am incredibly grateful to my parents, relatives, and friends for their moral support. In addition, I would like to express my gratitude to the other MSCSE program faculty members who have either directly or indirectly helped me in finishing this work by offering their invaluable support.

Finally, I would like to thank the dept. of CSE of UIU for providing me with the chance to work on my thesis and for helping me during the entire Master of Science program.

Publication List

Work relating to the research presented in this thesis has been submitted by the author in the following peer-reviewed journal:

Journal Articles

1. Md. Parvez Hossain, Ohidujjaman, Mohammad Shorif Uddin, Muhammad Nurul Huda, and Tetsuya Shimamura, **Recognition of Bangla and English Words in Bengali Texts Using a Modified BERT-base-NER Model**, *Iraqi Journal for Computer Science and Mathematics*, Vol. 6, No. 4, December 2025, pp. 5.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Nowadays, Bengali people are using a combination of English and Bangla words on social media frequently. Many common Bangla words are disappearing from our daily usage due to the frequent use of English words in Bangla texts [1]. Therefore, it is a remarkable challenge for the Bengali nation and the language. Table 1.1 shows sample sentences in daily usage where Bangla-English words mixed Bangla sentences are in tokenized form with their tags as well as the standard Bangla words of the corresponding English words. In this work, we mitigate such a challenging issue using the context of natural language processing.

Table 1.1: Illustration of Bangla text consisting of Bangla and English words

Input Text	আমরা	দৈনন্দিন	কাজে	অনেক	ইংলিশ	ওয়ার্ড	ইউজ	করি
Tags	bn	bn	bn	bn	en	en	en	bn
Output Text	আমরা	দৈনন্দিন	কাজে	অনেক	ইংরেজি	শব্দ	ব্যবহার	করি
Input Text	ডেডলাইন	চলে	এসেছে	ফাস্ট	কাজ	শেষ	করতে	হবে
Tags	en	bn	bn	en	bn	bn	bn	bn
Output Text	সময়সীমা	চলে	এসেছে	দ্রুত	কাজ	শেষ	করতে	হবে
Input Text	কেন্দ্রের	নির্দেশে	সীতাকুণ্ডের	রোড	মিটিং	বাতিল	করা	হয়েছে
Tags	bn	bn	ne	en	en	bn	bn	bn
Output Text	কেন্দ্রের	নির্দেশে	সীতাকুণ্ডের	সড়ক	সভা	বাতিল	করা	হয়েছে

In recent years, natural language processing (NLP) has improved significantly by launching a reliable model such as bidirectional encoder representations from transformers (BERT). The BERT model performs admirably on language identification [1–

3], named entity recognition (NER)[4–6], text classification, sentiment analysis [7–9], offensive language identification [10, 11], and so on. The NER technique identifies and classifies the names of individuals, locations, organizations, and so on from linguistic text [4–6]. The conventional BERT-base-NER model can recognize four types of entities: location, organization, person, and miscellaneous [12, 13].

1.1 Problem Domain and Motivation

There are cultural and practical implications to the growing practice of incorporating English words into Bangla. The original structure and richness of the Bengali language are undermined by the overuse of English words in Bangla, making it more difficult to preserve the language's purity and uniqueness. A vital component of cultural identity is language. Future generations may get detached from the traditional vocabulary and expressions that reflect Bengali heritage if Bangla is excessively influenced by English. Communication obstacles between various societal sectors may result from the inability of many people, particularly those living in rural areas, to comprehend conversations in mixed languages. The use and growth of the Bangla vocabulary may be slowed down over time when people replace English words for the original Bangla words shown in Table 1.1. As a result, distinctive Bangla words and expressions may become extinct. Overemphasizing English in Bangla has the risk that future generations prefer English over Bangla entirely, which would gradually reduce Bangla language usage and proficiency. Insignificant research is done on the identification of English words in Bangla text. Very limited corpus of Bangla-English code mixed data is available especially in Bangla text.

1.2 Objectives of the Thesis

- To construct a corpus of Bangla texts incorporating Bangla-English code-mixed data.
- To develop a system that can recognize English and Bangla words from Bangla texts.
- To translate the identified English words into standard Bangla words using the Google Translate API.

1.3 Thesis Contributions

To do this, we have modified the Bert-base-NER model and train the model with the proposed Bangla dataset, which consists of Bangla text samples with Bangla and English words. The proposed Bert-base-NER model's performance is assessed in order to recognize English and Bangla words from Bangla texts.

The study provides an in-depth analysis of the modified Bert-base-NER model's ability to recognize Bangla and English words from Bangla texts. In order to assess the effectiveness of the modified Bert-base-NER model and its potential for useful applications, we also compare its performance with that of other machine learning and deep learning models for the identification of Bangla and English words from Bangla texts. Here's a summary of our major contributions:

- We have created an extensive dataset that contains both English and Bangla words in Bangla texts.
- We have modified the Bert-base-NER model and evaluated the model with the proposed Bangla dataset.
- Additionally, we employed SVM, Naive Bayes, CNN+LSTM, and BiLSTM models training through the proposed Bangla dataset.
- Finally, The system converts English words into standard Bangla words using Google Translate API, enhancing the clarity and effectiveness of communication.

1.4 Organization of the Thesis

The thesis book is organized as follows:

Chapter 2 provides related works on language identification at the word level specially on code-mixed data.

Chapter 3 presents the proposed methodology including the data collection and dataset preparation process.

Chapter 4 explains the findings and experimental analysis of this study.

Chapter 5 outlines the findings, highlights the contributions made by the thesis, and discusses future work.

Chapter 2

Related Work

In this chapter, we studied extensively in several journal articles and conference papers on language identification on code-mixed data and named entity recognition (NER) using machine learning, deep learning and transformer models. We have tried to analyze each work with their features and limitations in some cases. We have also discussed about conventional Bert-base-NER model and it's work process.

2.1 Related Works

In natural language processing, identifying individual linguistic terms between Bangla and English while writing on social media is frequently overlooked. Borhan *et al.* address the impact of the widespread use of Banglish words (simply English words written in Bangla script) on the quality of Bangla writing texts [1]. In [1], the study suggests a reliable method for identifying and converting Banglish words (simply English words written in Bangla script) into standard Bangla words using the NER-BERT model with an overall accuracy of 80%. In their work, they utilized a Dataset contains 700 sentences only. They have utilized a self made dictionary for Banglish to standard bangla conversion. If there are no Banglish terms in the dictionary, the system does not convert them, which is one limitation of their work.

Hidayatullah *et al.* proposed a framework for code-mixed language identification (LID) and identified four important factors: techniques, challenges, data availability, and quality criteria [14]. Following the verification purpose, 32 code-mixed corpuses are prepared that are available to identify code-mixed languages. In order to address

code-mixed language identification (LID) issues, they found gaps and potential areas for further research. They identified four major issues: intra-word code-mixing, lexical borrowing, ambiguity, and non-standard terms. Results showed that the multichannel CNN combined with CRF and BiLSTM has demonstrated exceptional result in resolving code-mixed LID difficulties in certain neural network-based investigations. Regarding the non-neural network methods, it is advised to use CRF and SVM. The transformed-based approach is among the most reliable methods for code-mixed LID because of its exceptional performance.

A system for identifying Kannada-English code-mixed data on the CoLI-Kanglish corpus was proposed by Balouchzahi *et al.* [15] and consisted of six criteria: English (en), Kannada (kn), Kannada-English (kn-en), Location (location), Name (name), and Other (other). The averaged weighted and averaged macro F1 scores have been taken into consideration as the evaluation measures. The top-performing model among the ones that participants submitted had F1 scores of 0.62 (macro) and 0.86 (weighted).

Thara *et al.* focus in Malayalam-English code-mixed word-level language identification (WLLI) on websites like YouTube [16]. To identify languages at the word level, they employ the transformer model BERT and its variations, CamemBERT and DistilBERT. Six labels English, Malayalam, universal, mixed, acronyms, and undefined are applied to Malayalam-English code-mixed data in the suggested method. A brand-new Malayalam-English corpus was made in order to assess the performance of cutting-edge models such as BERT. Only word-level language identification has been done in their study; code-mixed words have not been converted to standard terms.

Gundapu *et al.* suggested a study involving multiple models, namely random forest (RF) regression, the naive Bayes classifier, the hidden Markov model (HMM), and the conditional random field (CRF) to mitigate the problem of language identification in the English-Telugu Code-Mixed dataset [17]. Among them, the CRF-based model performs more suitably with an F1-score of 0.91 than the others.

Chaitanya *et al.* look into language identification in the context of code-mixed social media data, with a particular emphasis on Facebook, which as of the first quarter of 2018 had over 2.19 billion users [18]. To create feature vectors, the methodology compares different word embedding techniques, such as Continuous Bag of Words (CBOW) and Skip-Gram models. Then, several ML algorithms, KNN (K-nearest neighbors),

SVM, LR(logistic regression), RF, GNB (Gaussian naive Bayes), and Adaboost used feature vectors as inputs. The cross-validation scores show promise in the results.

Das *et al.* used both character-based and phonetic-based word encoding approaches to train their deep LSTM models [19]. With the help of these two models, they were able to use the stacking and threshold procedures to generate two ensemble models with respective accuracy on testing data of 91.78% and 92.35%.

2.2 Conventional Methods

In recent years, most tasks involving named entity recognition and word-level language identification have been completed by machine learning, deep learning, and BERT base transformer models. However, BERT has performed better than other prevailing models in word-level language identification and named entity recognition [16].

2.2.1 Bert-base-NER Model

Bert-base-NER is a fine-tuned BERT model created especially for named entity recognition tasks. It can be used for text classification, natural language understanding, language modeling, and information extraction [12]. Bert-base-NER model follows the standard transformer-based token classification framework.
For Input Embedding, each input token is represented as:

$$E(x_i) = T_i + S_i + P_i \tag{2.1}$$

In Eq. 2.1, T_i represents the token embedding, S_i represents segment embedding, and P_i represents positional embedding. The Bert-base-NER model learns token representations using BERT embeddings. For Self-Attention Mechanism, the attention mechanism computes the attention score as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.2}$$

In Eq. 2.2, $Q = XW_Q$, $K = XW_K$, $V = XW_V$ (learnable parameter matrices) and d_k represents Dimension of key vectors. Self-attention helps capture contextual relationships. For Feedforward Network, the output from the self-attention layer is processed through a feedforward network:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{2.3}$$

In Eq. 2.3, W_1, W_2, b_1, and b_2 are trainable parameters. A feedforward network refines token features. For Token Classification, for Named Entity Recognition, each token's final hidden state is classified as:

$$y_i = \text{softmax}(Wh_i + b) \tag{2.4}$$

In Eq. 2.4, y_i represents the probability distribution over entity classes. A softmax classifier predicts one of four classes: Name, Place, Organization, and Miscellaneous. For Loss Function, the model is optimized using categorical cross-entropy loss:

$$\mathcal{L} = -\sum_i \sum_j y_{i,j} \log(\hat{y}_{i,j}) \tag{2.5}$$

In Eq. 2.5, $y_{i,j}$ is the true one-hot encoded label and $\hat{y}_{i,j}$ is the predicted probability for class j. The cross-entropy loss function ensures optimal classification.

2.3 State-of-the-Art Techniques

This section discusses the most recent developments and approaches in word level language identification and named entity recognition, emphasizing those that have demonstrated potential for use in a range of text-processing applications.

Table 2.1: Research on Word Level Language Identification

Reference	Summary
[1]	Proposes a reliable approach for detecting Banglish words written in Bangla script, utilizing computational techniques to handle the constraints of mixed-language text analysis.
[3]	The study proposes a transformer-based model for word-level language recognition in Kannada-English code-mixed texts, proving the utility of deep learning for multilingual text processing.
[15]	This study presents an overview of CoLI-Kanglish, a shared task focused on language identification at word-level in Kannada-English code-mixed texts at ICON 2022, emphasizing methodologies, obstacles, and system performances.
[16]	This work presents a transformer-based methodology for language detection in Malayalam-English code-mixed text, demonstrating robust performance on multilingual and informal text datasets.
[19]	This research presents character- and phonetic-based LSTM models for language recognition in Bangla-English code-mixed data, successfully identifying linguistic patterns for enhanced classification.
[20]	In order to overcome the difficulties of having few datasets and enhance identification accuracy, this work suggests a word-level language identification technique designed for languages with minimal resources.
[21]	Provides a word-level language recognition method for social media material that is code-mixed in Assamese, Bangla, Hindi, and English, addressing the challenges of informal and multilingual communication.

Table 2.2: Research on Named Entity Recognition

Reference	Summary
[4]	In order to improve entity detection in noisy, multilingual search inputs, this work presents a gazetteer-enhanced method for named entity recognition for code-mixed search queries.
[5]	proposes a meta-embedding-based method that integrates various embedding representations to improve performance for named entity recognition in code-mixed Indian language corpora.
[6]	The study uses machine learning approaches to overcome the difficulties of multilingual and mixed-script data by presenting a Named Entity Recognition (NER) system designed for Arabic-English code-mixed text.
[12]	Through transfer learning, the paper suggests Bangla-BERT, a transformer-based language model optimized for Bangla that shows good results on a range of natural language understanding tasks.
[13]	Introduces BanglaBERT, a Bangla masked language model created to enhance Bangla natural language comprehension challenges. It was trained on extensive Bangla corpora.

2.4 Summary

Some other work has also been done on word-level language identification [20–25]. Among the previous studies, rule-based or shallow learning techniques are implemented for code-mixed language identification [17–19]. Commonly, formal evaluation criteria are absent in the rule-based learning [1, 24]. High-quality annotated datasets for Bangla-English code-mixed text data are scarce [14]. Transformer-based models (e.g. BERT) provided effectiveness in multilingual contexts [2, 3]. No prior work was performed to modify BanglaBERT on Bangla-script code-mixed data [12, 13]. Therefore, there is enough scope to improve the BERT-base-NER model with Bangla script code-mixed data. To address data set limitation, we developed a word-level annotated corpus of Bangla-English code-mixed text data from social networks. In this paper, we measure the reliability of the annotation using Cohen's Kappa to ensure consistency. We train ML, DL, and modified BERT-base-NER models using the training Bangla-English code-mixed data. The modified BERT-base-NER model performs state-of-the-art for Bangla (bn), English (en), Name Entity (ne), Unknown Entity(un) recognition purposes. In addition, we propose an algorithm integrating the Google Translate API to convert detected English words to standard Bangla.

Chapter 3

Proposed Methodology

In this chapter, we have discussed the data collection and dataset preparation process. After that different ML and DL models such as Naive Bayes, SVM, CNN+LSTM, and BiLSTM were trained to predict word tags from Bangla texts. Besides we have also trained with a modified Bert-base-NER model which has better tag accuracy. Fig. 3.2 shows the flow diagram of our proposed system.

3.1 Data Collection and Dataset Preparation

The extraction of Raw text from social media networks is a significant component of the creation of annotated corpora. From a range of websites, including Facebook, YouTube, and Instagram, we collected sentences. The sentences included some self written sentences. With the addition of a CSV file, we are compiling a dataset of 1742 sentences that use various Bangla and English words. Each Sentence has its own ID. Our data is then forwarded to the preprocessing stage. Sentences are tokenized into words with similar sentence IDs after preprocessing. Next, we have manually annotated each word with its language identification tag. When annotating, we employed four different kinds of tags. These are UN (for Others), NE (for Named Entities), EN (for English Words), and BN (for Bangla Words). Table 3.1 shows the sample of our Corpus. And Table 3.2 and Fig. 3.1 display the Corpus's statistics. We have considered name, place and organization as Name entity in this work.

Inter-Annotator Agreement (IAA) quantifies the consistency of annotations provided by multiple annotators engaged in the same task or dataset during the creation

Table 3.1: Bangla Texts Dataset Consisting Mixing of Bangla and English Words

sentence_id	words	labels
0	কেন্দ্রের	be
0	নির্দেশে	be
0	সীতাকুণ্ডের	ne
0	রোড	en
0	মিটিং	en
0	বাতিল	bn
1	সরকারের	bn
1	হুমকিতেই	bn
1	দেশ	bn
1	ছেড়েছি	bn
1	সুরেন্দ্র	ne
1	সিনহা	ne

Table 3.2: Statics of Bangla Texts Dataset Consisting Mixing of Bangla and English Words

Language Label	Label Frequency
bn	7179
en	2209
ne	667
un	434

of the training dataset. We employed Cohen's Kappa for IAA for data annotation as a significant statistic for assessing annotation consistency and reliability. We have a Kappa value of 0.99, which shows strong agreement beyond chance and so reliable annotations. We utilized 1393 sentences with 8325 words for training and 349 sentences with 2162 words for testing purposes. Additionally we have applied 5-fold cross validation to generalize the proposed model.

20% of the total data is set aside for testing, and the remaining data is utilized to train the model.

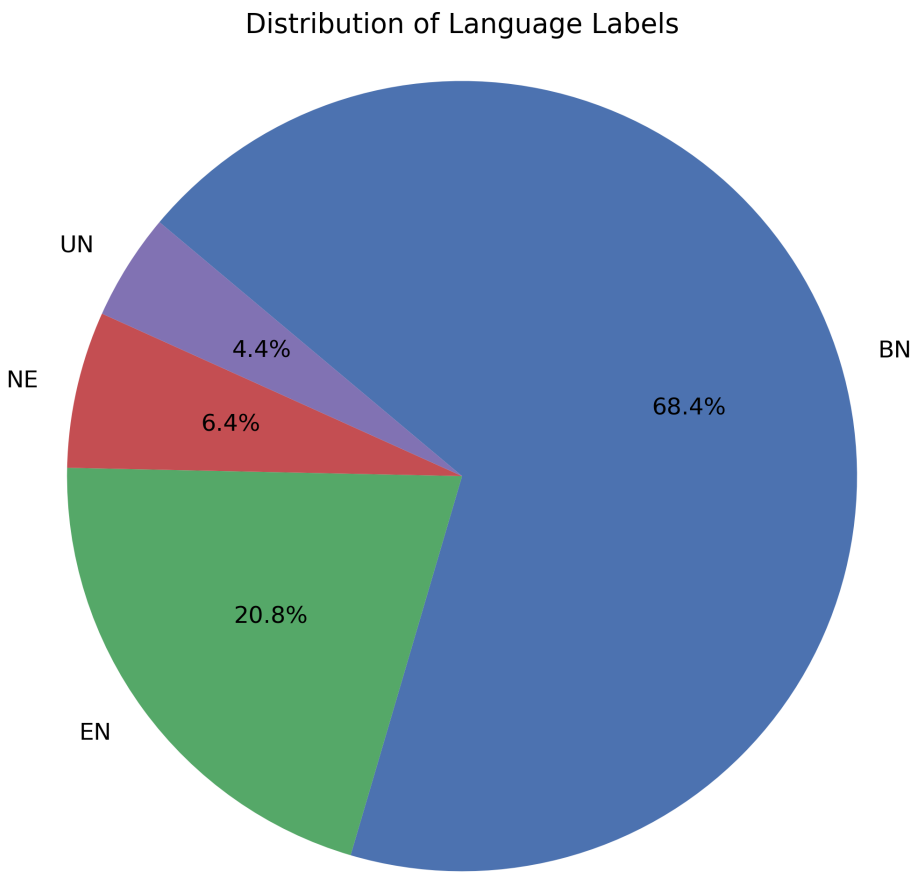

Figure 3.1: Statistics of different language tags

3.1.1 Data preprocessing

Data preparation is an essential step in every study. As a result, the original dataset is gathered and carefully cleaned. By grouping sentences together and employing token values, duplicate sentences are removed. Following the dataset cleansing procedure, each sentence is parsed separately in order to give each one a unique identifier (ID).

Using the pandas library, the parsed statements are further divided into words. Because a single phrase is made up of several words, dissecting it yields multiple null values in the sentence ID. This is taken care of in the preprocessing step by using a label encoder to fill in null values.

Finally we have got 10487 words from 1742 sentences. Next, we manually added the language identification tag to each word. We used four different types of tags when

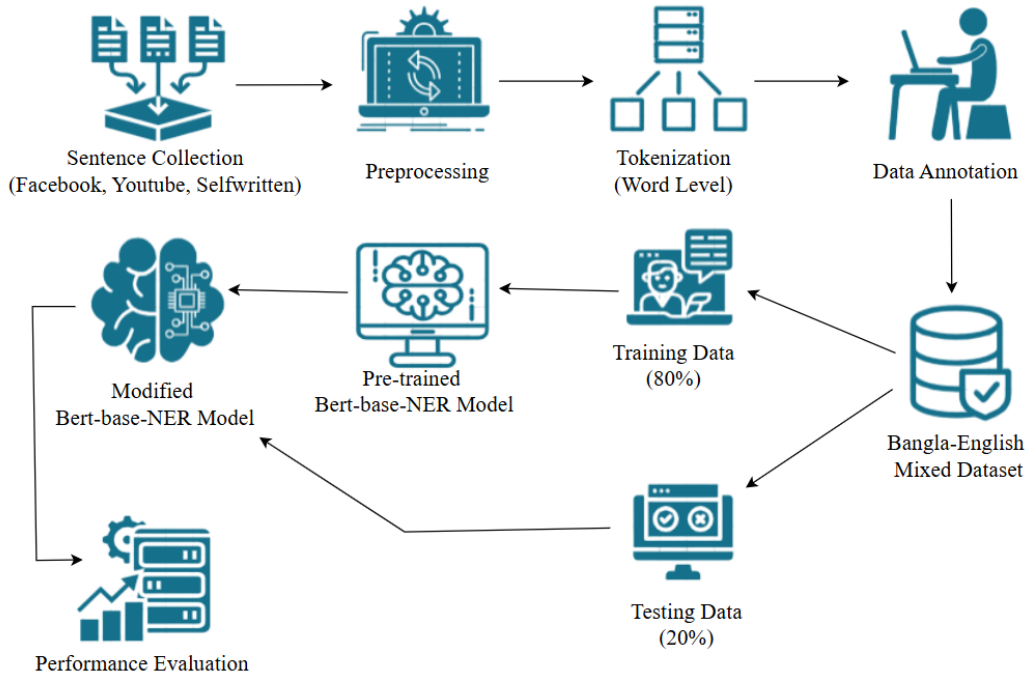

Figure 3.2: Framework of Model Development for English Words Identification using Input Dataset

annotating. These are BN (for Bangla Words), EN (for English Words), NE (for Named Entities), and UN (for Others) which is already discussed in section 3.1.

3.2 Modified Bert-base-NER Model

BERT is based on the Transformer architecture, which uses an attention mechanism to figure out the word-context relationships within a text [26, 27]. An encoder and decoder are used for reading input text and task prediction, respectively. An encoder and a decoder form a basic transformer. However, an encoder component is necessary because BERT is designed to generate a linguistic representation model. A series of tokens is processed by the BERT encoder, which converts the input into contextualized embeddings. We redesigned the original BERT-base-NER model by adding a BiLSTM layer on top of these embeddings. This enhancement allows the model to more effectively capture sequential patterns and relationships in the input text, improving its capacity to reliably identify named entities by integrating BERT's profound contextual comprehension with explicit sequence modeling. Afterward, the output vectors are fed

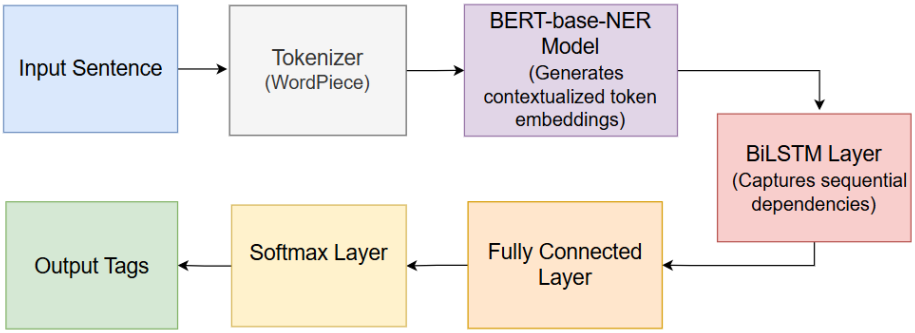

Figure 3.3: Structure of modified BERT-base-NER model

into a task-specific fully connected layer followed by a Softmax function to manage the multiclass classification. The modified BERT-base-NER model is shown in Figure 3.3. Before processing, the input of BERT needs to be improved and modified in certain ways with extra metadata. The certain ways are discussed as follows:

3.2.1 Token Embeddings

As part of the input word tokens, a special [CLS] token is added at the beginning of the first sentence, followed by a [SEP] token at the end of each sentence.

3.2.2 Positional Embeddings

Positional embeddings are assigned to each token to indicate its location within the text. This aids in the model's comprehension of the words' sequential order.

3.2.3 Segment Embeddings

Sentence 0 or Sentence 1 is indicated by the segment embedding that is attached to each token. The encoder can distinguish between distinct sentences thanks to this segmentation.

The embeddings process is shown in Fig. 3.4. The BERT encoder is primed to process the input sequence efficiently with some changes, such as i) tokenization, ii) adding special tokens (CLS, SEP), iii) segment & position embeddings, and iv) padding and attention masks. Token, positional, and segment embeddings are combined to enhance the model's comprehension of the sequential and contextual aspects of the input text. Thus, it strengthens BERT's language representation capabilities. The

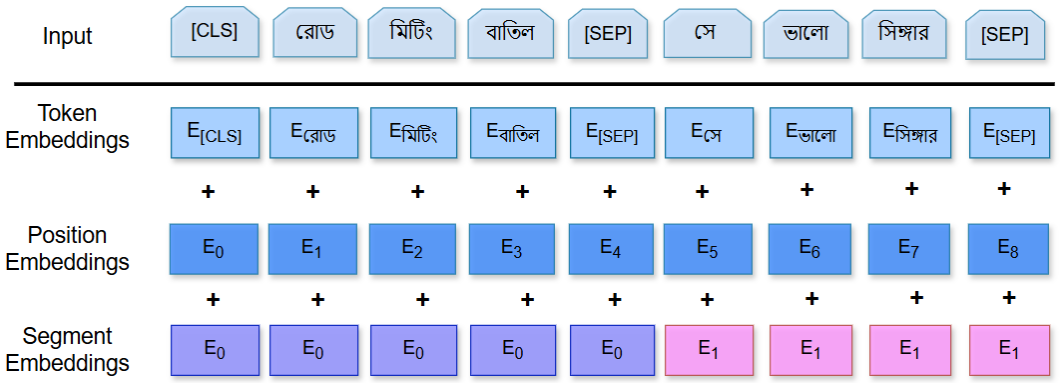

Figure 3.4: BERT Embedding for Our System

transformer stacks a layer that maps a sequence to sequence, resulting in an output that is likewise a sequence of vectors at the same index, where input and output tokens are in one-to-one correspondence. Table 3.3 shows the comparative analysis between the conventional and BERT-base-NER model and the proposed BERT-base-NER model. The flow diagram of the proposed system to recognize Bangla and English words and convert English words into standard Bangla words is shown in Fig. 3.5. The proposed system takes a sentence having Bangla or English words in Bangla texts as input to preprocess the input text. In the preprocessing part, the punctuation and special characters are removed. Sentences are tokenized into words after the preprocessing. The proposed BERT-base-NER model predicts and assigns language tags using the Algorithm 1. An instance of words prediction is shown in Table. 3.4.

3.3 Conversion of English to Standard Bangla Words

The proposed system converts English words into standard Bangla words after the recognition of English and Bangla words. However, a tag is found as **en**, which means an English word, and the system replaces the English word with a standard Bangla word using the Google Translate API. The conversion process from English to standard Bangla is performed using the Algorithm 2. Finally, we obtain the standard Bangla sentence as an output. Table 3.5 shows an example of translating English into standard Bangla words using the Google Translate API.

Algorithm 1 Word Level Language Tagging

Input ← $X(Bangla\_and\_English\_mixed\_Bangla\_Sentence)$
Tokenize the sentence using the BERT tokenizer
Run the Bert_base_NER model on the tokenized sentence
for *each word in X* **do**
    if *entity label belongs to Bangla* **then**
        $tag.append(bn)$
    end if
    if *entity label belongs to English* **then**
        $tag.append(en)$
    end if
    if *entity label belongs to Named_Entity* **then**
        $tag.append(ne)$
    end if
    if *entity label belongs to Unknown language* **then**
        $tag.append(un)$
    end if
end for
Output ← $Y(Words\_with\_Tags)$

Algorithm 2 Conversion of English Words into Standard Bangla Words

Input ← $X(Bangla\_and\_English\_Words\_with\_Tags)$
Integrate the Google_Translator_API
Create an Empty List l
for $w \leftarrow$ *Each word in X* **do**
    if (w_tag == "en") **then**
        $e \leftarrow$ *Translate w into English*
        $b \leftarrow$ *Translate e into Bangla*
        $l.append(b)$
    else
        $l.append(w)$
    end if
end for
Output ← $Y(Standard\_Bangla\_Sentence)$

Table 3.3: Comparative analysis between the conventional BERT-base-NER model and the proposed model

Context	**Conventional BERT-base-NER**	**Modified BERT-base-NER**
purpose	named entity recognition	word-level language identification in Bangali text
target dataset	standard NER dataset	customized Bangla-English code-mixed corpus
class	person, location, organization, others	bn, en , ne , un
lingo support	monolingual (commonly English)	code-mixed Bangla-English
script	Roman text script	Bangla text script
adaptability to social media	low (trained on formal text)	high (fine-tuned on noise, informal social media)
fine-tuned dataset	formal NER dataset	newly annotated code-mixed corpus with inter-annotator agreement

Table 3.4: Prediction of different tags

Input words	Prediction tags
পারভেজ	ne
তোমাকে	bn
অনেক	bn
প্রবলেম	en
সলভ	en
করতে	bn
হবে	bn

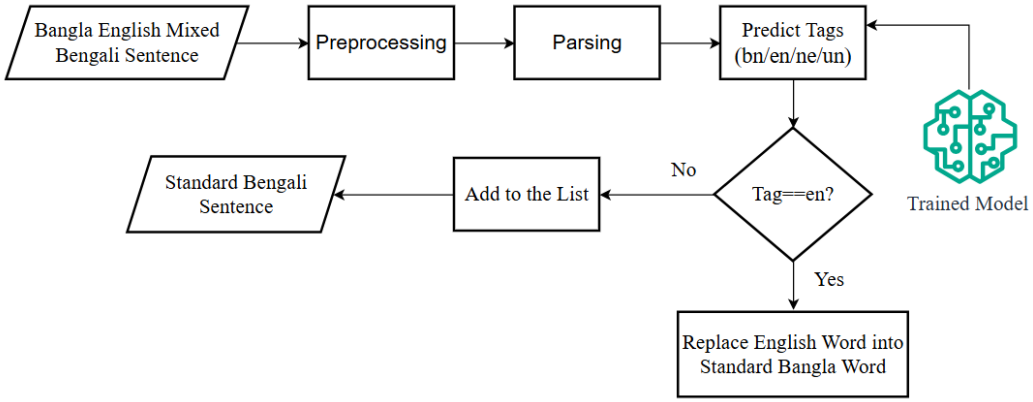

Figure 3.5: Flow Diagram of the Proposed System to Identify English Words from Bangla texts and Convert them into Standard Bangla Words

Table 3.5: Process of English to Standard Bangla Word Conversion Using Google Translator API

Bangla Words	English Words	Standard Bangla Words
রোড	Road	সড়ক
মিটিং	Meeting	সভা
সিচুয়েশন	Situation	পরিস্থিতি
অফিস	Office	দপ্তর

3.4 Summary

A method for identifying Bangla and English words in Bangla texts and converting English words into standard Bangla words is presented in the study. The two primary components of the system are the recognition of words in Bangla and English and the conversion of English words into standard Bangla words. To predict the tag from the input text, a modified Bert-base-NER model is proposed. And to translate English words into conventional Bangla words, the Google Translator API is used.

Chapter 4

Experimental Analysis

4.1 Experimental Setup

The Google Colab system used for the experiment had an Intel(R) Core(TM) i5-6300U CPU running at 2.40GHz and 2.50GHz, as well as Tesla T4 GPUs, each with 8 GB of RAM. When implementing the model, two important libraries were utilized: PyTorch and TensorFlow. This configuration offered strong processing power for effective transformer model training. Our experiments' hyperparameters are shown in Table 4.1, which offers information on the configuration settings that guided the training procedure.

Table 4.1: Hyperparameter values for proposed Bert-base-NER model

Serial No	**Parameter/Technique**	**Details**
1	Learning Rate	$2e-5$
2	Number of Epochs	5
3	Batch Size	32
4	Max Length	128
5	Weight Decay	0.01
6	Class Mode	`multclass`
7	Metrics	`accuracy`
8	Dropout Rate	0.2

4.2 Evaluation Metrics

The primary evaluation metrics such as precision, recall, F1-score, accuracy, error rate, specificity, and ROC curve. These metrics will shed light on how well the model detects Bangla and English words from Bangla texts while minimizing false positives and false negatives.

$$Precision = \frac{TP}{TP + FP} \times 100 \tag{4.1}$$

Eq. 4.1 represents the equation for Precision, It assists in determining what percentage of affirmative identifications are correct.

$$Recall = \frac{TP}{TP + FN} \times 100 \tag{4.2}$$

Recall in the equation 4.2, which explains the proportion of true positives, was successfully identified.

$$F - measure = \frac{2 * precision * recall}{precision + recall} \tag{4.3}$$

Using Eq. 4.3 F-Measure is calculated which is the harmonic mean of precision and recall, providing a balance between the two metrics.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.4}$$

By using Eq. 4.4 to calculate the ratio of accurately predicted instances (both positive and negative) to the total number of predictions, accuracy measures the model's overall correctness.

The Error Rate is defined as the proportion of all incorrect predictions out of the total number of predictions calculated by Eq. 4.5.

$$\text{Error Rate} = \frac{FP + FN}{TP + TN + FP + FN} \tag{4.5}$$

Specificity measures the proportion of predictions with a negative label that the model correctly classifies. It is commonly called the true positive ratio (TPR). Specificity is defined as follows,

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{4.6}$$

Training accuracy and loss: In Fig. 4.1, training accuracy and loss curves are plotted using the proposed method over 5 epochs. The accuracy curve illustrates how the training accuracy increases throughout epochs to surpass 93%, while the validation accuracy

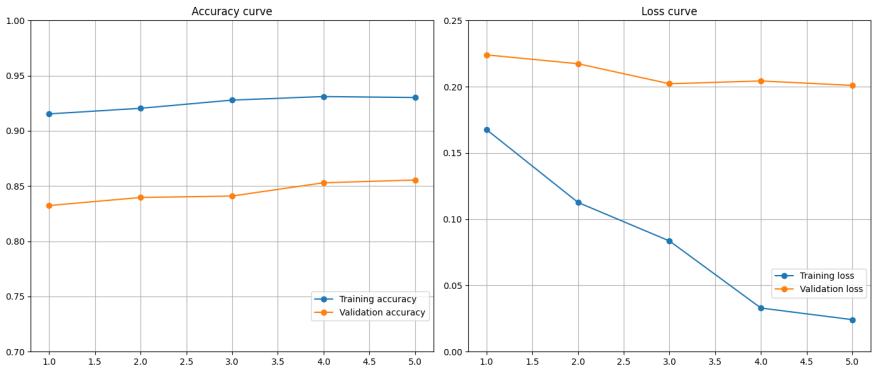

Figure 4.1: Training accuracy and loss curve

also rises, reaching a peak of approximately 85.5% by the end of the epoch. This phenomenon suggests a steady capacity for learning and generalization. The tendency is further supported by the loss curve. The validation loss remains reasonably consistent following an initial drop, indicating that the model does not overfit and continues to perform well on unseen data. In contrast, the training loss drops dramatically with each epoch, indicating good optimization.

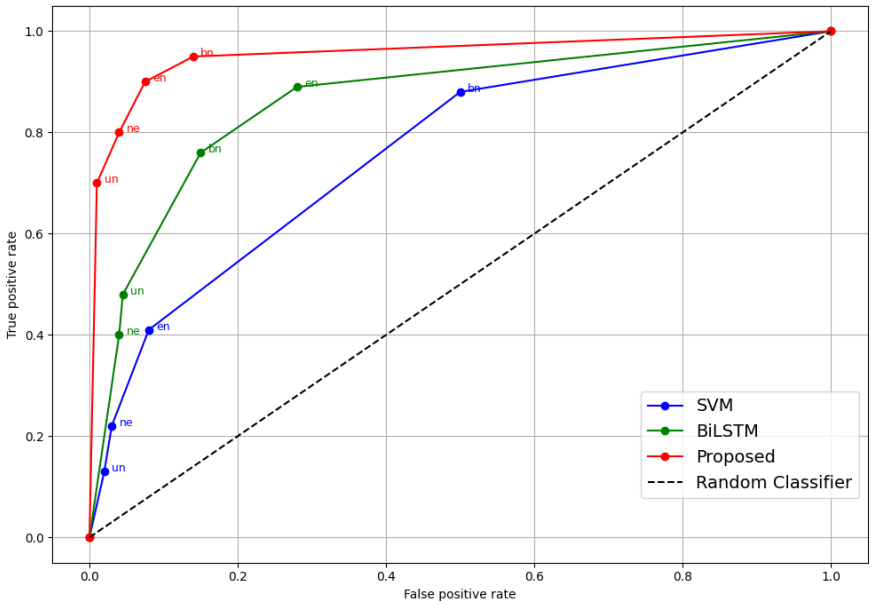

Figure 4.2: ROC curve for all models

The ROC (receiver operating characteristic) curve is a powerful tool for assessing

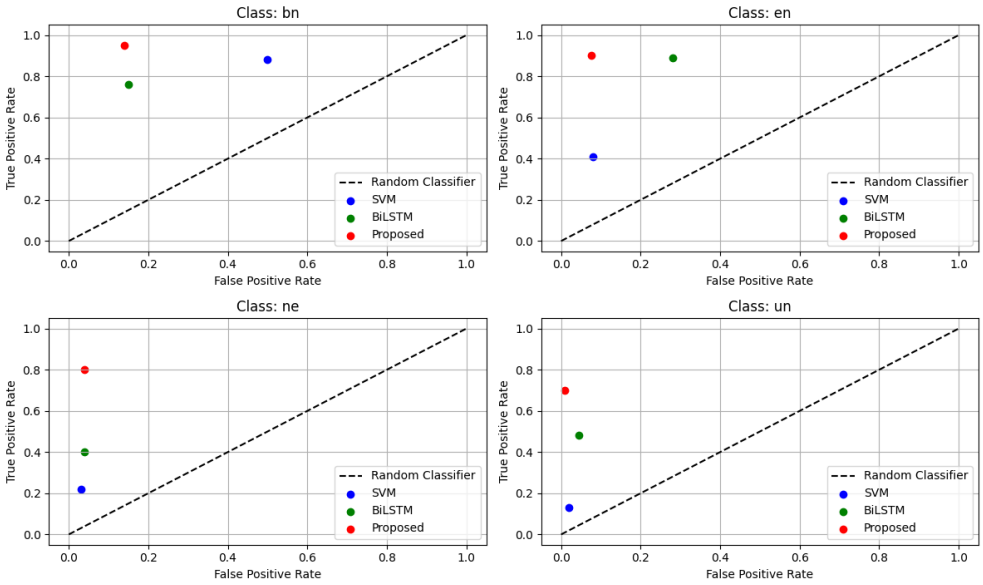

Figure 4.3: Class-wise ROC comparison of different models

and comparing the performance of various methods. The ROC curve is sketched by estimating the true positive rate (TPR) and false positive rate (FPR) at different thresholds. We can obtain the ROC curve by plotting FPR and TPR on the X-axis and Y-axis, respectively. Each point on the curve represents a specific decision threshold as shown in Fig. 4.2. The TPR and FPR are expressed as,

$$\text{TPR} = \frac{TP}{TP + FN} \tag{4.7}$$

$$\text{FPR} = \frac{FP}{FP + TN} \tag{4.8}$$

We generate a class-wise ROC curve as shown in Fig.4.3, which shows the effectiveness of three models, such as SVM, BiLSTM, and the proposed method, across four classes: bn, en, ne, and ue. Each subplot displays the true positive rate and false positive rate for a particular class. The proposed model continuously obtains higher TPR with lower FPR in contrast to SVM and BiLSTM, indicating greater achievement potential. The dashed diagonal line denotes the effectiveness of the random classifier. In Fig. 4.4, a single class performance is represented by each point on the curves. Across all classes, the proposed model provides suitable classification performance since it gets closer to the optimal top-left corner of the ROC space. In Fig. 4.4, we note that the proposed method provides an ROC curve over the baseline methods.

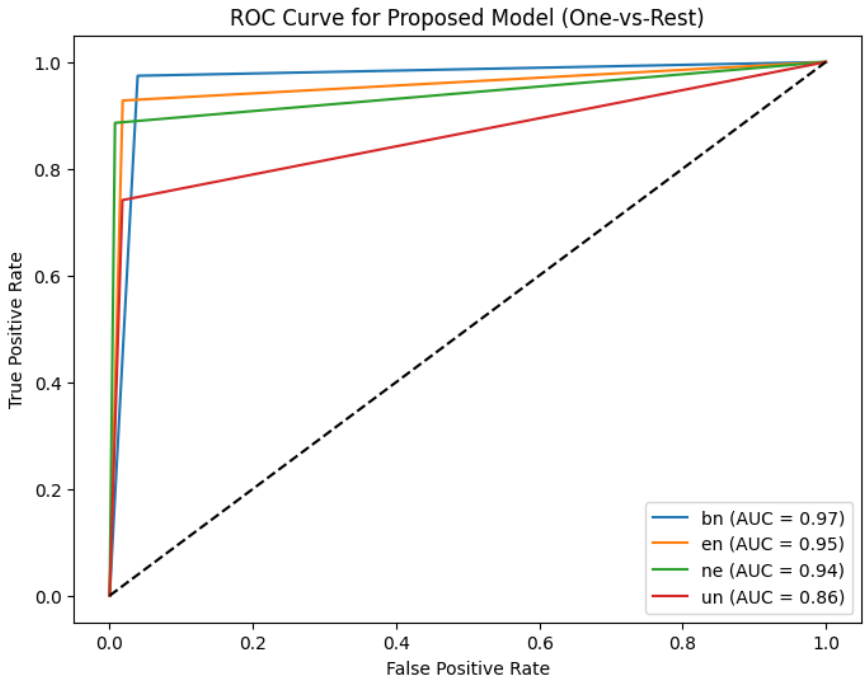

Figure 4.4: ROC curve for proposed Bert-base-NER model

4.3 Results Discussion

We have generated confusion matrix for all the models we experimented on our dataset. Fig. 4.5 shows the confusion matrix for SVM model, Fig. 4.6 shows the confusion matrix for Naive Bayes, Fig. 4.7 shows the confusion matrix for CNN+LSTM, Fig. 4.8 shows the confusion matrix for BiLSTM, and Fig. 4.9 shows the confusion matrix for our proposed model. From Fig. 4.5 and 4.6, we can see that bn and en class perform better than ne and un class due to lower data of that classes. On the other hand, from Fig 4.7 and 4.8, we can see that class ne and un perform better.

Fig. 4.10 shows that the proposed BERT-base-NER model can predict the tag of each word from the input sentence after implementing Algorithm 1. After the prediction, system converts the english words (having **en** tags) into standard Bangla words using google translator api shown in Fig. 4.11. Table 4.2, 4.3, 4.4, 4.5, and 4.6 shows the class-wise precision, recall, f1 score and support for SVM, Naive Bayes, CNN + LSTM, BiLSTM and proposed model, respectively. From Table 4.2 and 4.3, we can see that bn and en class perform better than ne and un class due to lower data of that

Table 4.2: Results for SVM

class	precision	recall	f1-score	support
bn	0.77	0.98	0.86	1440
en	0.76	0.39	0.51	430
ne	0.69	0.18	0.28	140
un	0.50	0.05	0.08	88
accuracy		0.77		
weighted avg	0.76	0.77	0.73	2098

Table 4.3: Results for Naive Bayes

class	precision	recall	f1-score	support
bn	0.77	0.99	0.86	1440
en	0.74	0.40	0.52	430
ne	0.89	0.12	0.21	140
un	1.00	0.01	0.02	88
accuracy		0.76		
weighted avg	0.73	0.76	0.71	2098

Table 4.4: Results for CNN+LSTM

class	precision	recall	f1-score	support
bn	1.00	0.76	0.86	1471
en	0.97	0.56	0.71	457
ne	0.20	0.97	0.33	149
un	0.95	0.47	0.63	85
accuracy		0.72		
weighted avg	0.88	0.72	0.78	2162

Table 4.5: Results for BiLSTM

class	precision	recall	f1-score	support
bn	1.00	0.76	0.86	1471
en	0.49	0.99	0.65	457
ne	0.92	0.40	0.55	149
un	0.95	0.48	0.64	85
accuracy		0.85		
weighted avg	0.88	0.85	0.83	2162

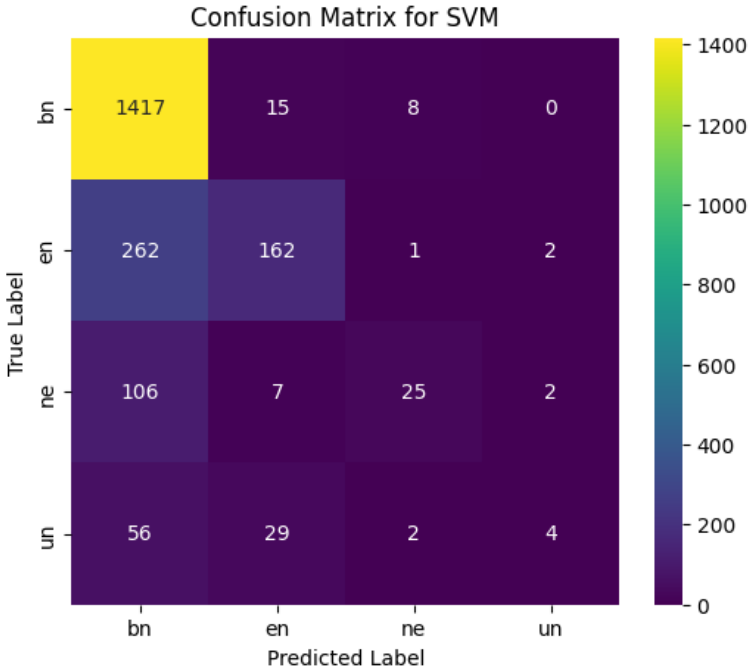

Figure 4.5: Confusion matrix for SVM

classes. On the other hand, from Table 4.4 and 4.5, we can see that class ne and un perform better. Table 4.7 shows the overall precision, recall, f1-score, and specificity of proposed model comparing with prevailing models. Besides hold-out approach, we have also applied 5 fold cross validation in the proposed model. It ensures that the model generalizes adequately to previously unseen data by training and testing on different chunks of the dataset in repeated iterations. The results shown in Table 4.8. We also applied another applicable dataset (Borhan [1]) on the proposed model. Results of this

Table 4.6: Results for proposed model

class	precision	recall	f1-score	support
bn	0.97	0.97	0.97	1471
en	0.90	0.92	0.91	457
ne	0.83	0.87	0.85	149
un	0.68	0.62	0.65	85
accuracy			0.95	
weighted avg	0.94	0.94	0.94	2162

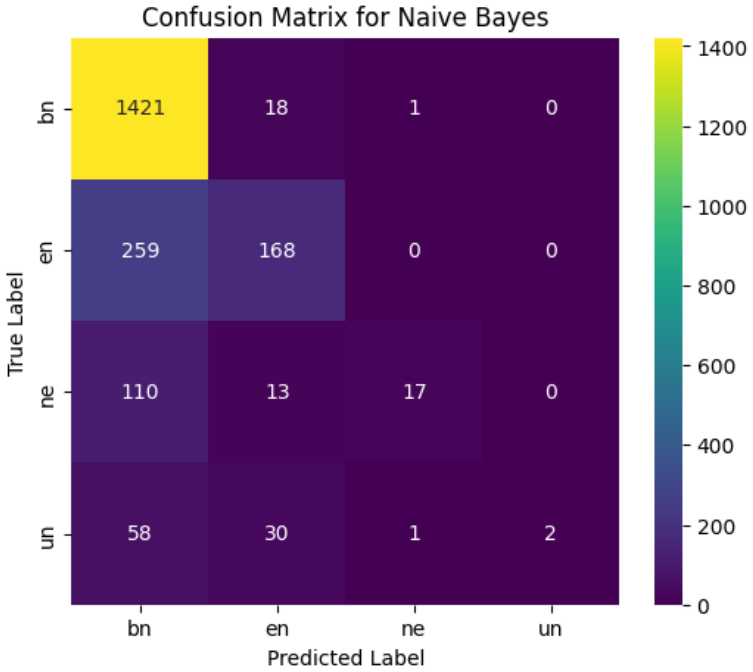

Figure 4.6: Confusion matrix for Naive Bayes

new dataset are shown in Table 4.9.

From 4.9 and 4.4 we can see that we achieved a good accuracy in BN, EN, and NE. For UN class, system has less accuracy than other class.

4.4 Comparison

We have found a significant amount of work has been done on code-mixed data, especially for word-level language identification. Almost all the works are in English

Table 4.7: Experimental results

Model	Precision	Recall	F1-score	Specificity
SVM	0.76	0.77	0.73	0.84
Naive Bayes	0.73	0.76	0.71	0.83
CNN+LSTM	0.88	0.72	0.78	0.92
BiLSTM	0.88	0.85	0.83	0.93
Proposed model	**0.94**	**0.94**	**0.94**	**0.98**

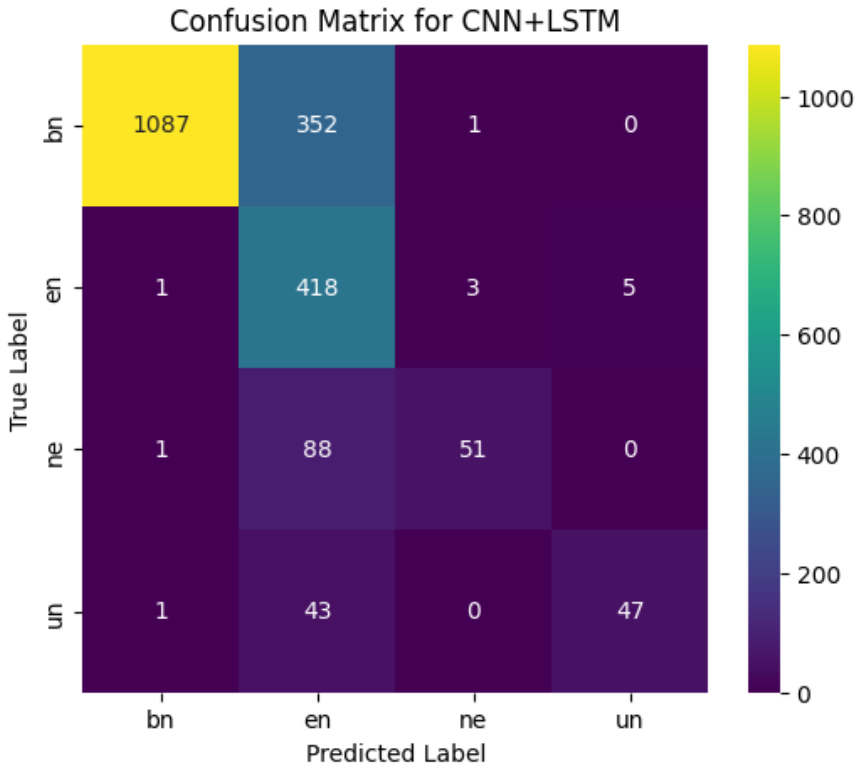

Figure 4.7: Confusion matrix for CNN+LSTM

Table 4.8: Model accuracy and error rate over 5 folds cross validation

Fold	**Accuracy**	**Error rate**
Fold-1	94.81%	5.19%
Fold-2	94.59%	5.41%
Fold-3	95.24%	5.76%
Fold-4	95.19%	5.81%
Fold-5	94.45%	5.55%
Average	94.86%	5.54%

and other language texts. We have found [1] is the only work on Bangla texts having Bangla and English words. Table 4.11 shows the accuracy comparison of our proposed Bert-base-NER model with other models.

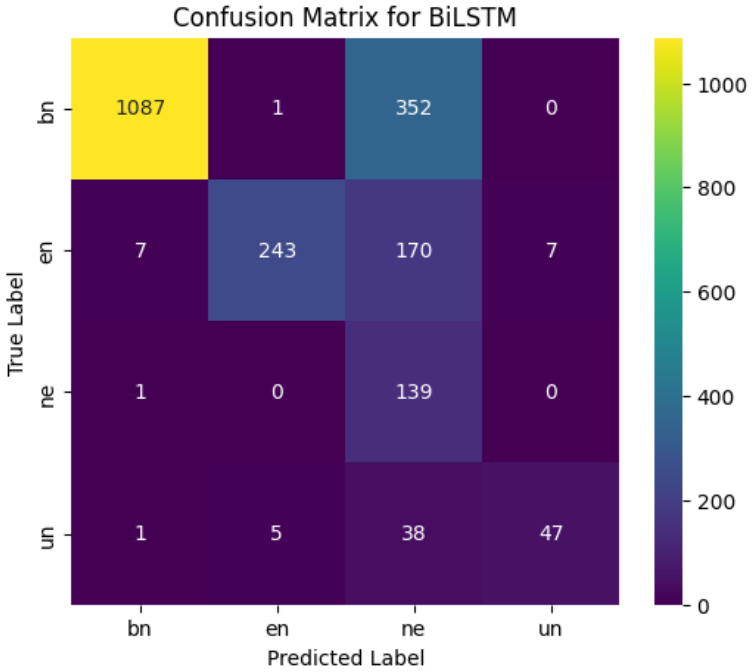

Figure 4.8: Confusion matrix for BiLSTM

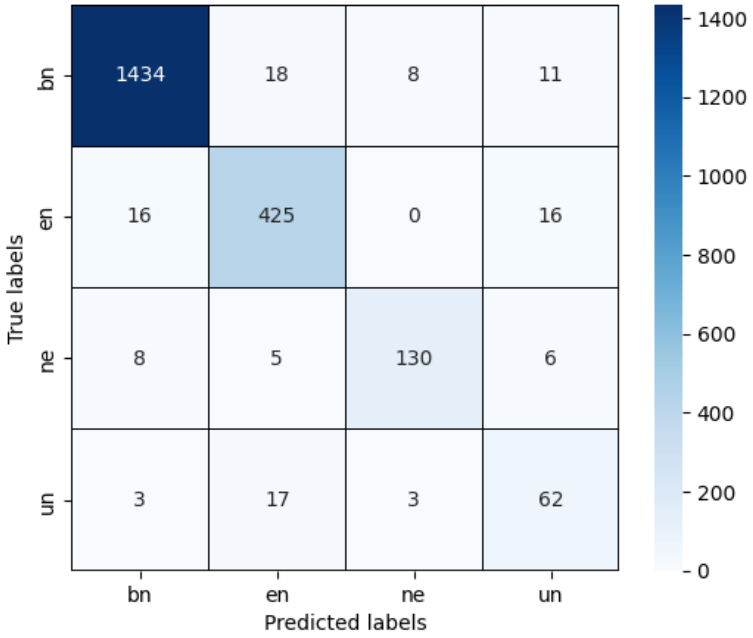

Figure 4.9: Confusion matrix for proposed Bert-base-NER model

Table 4.9: Model Performance Comparison with Similar dataset

Metric	Proposed Dataset	Borhan [1]
Precision	0.94	0.86
Recall	0.94	0.87
F1-score	0.94	0.86
Accuracy	0.95	0.87

Table 4.10: Accuracy over 5 folds using modified BERT-base-NER model and baseline BiLSTM model

Fold	Modified BERT-base-NER model	BiLSTM model
Fold-1	94.81%	85.77%
Fold-2	94.59%	84.59%
Fold-3	95.24%	85.24%
Fold-4	95.19%	84.00%
Fold-5	94.45%	84.45%
Average	94.86%	84.81%

Table 4.11: Model accuracy and error rate

Method	Accuracy	Error rate
CNN+LSTM	72%	28%
Naive Bayes	76%	24%
SVM	77%	23%
Borhan *et al.* [1]	80%	20%
BiLSTM	85%	15%
Proposed model	**95%**	**05%**

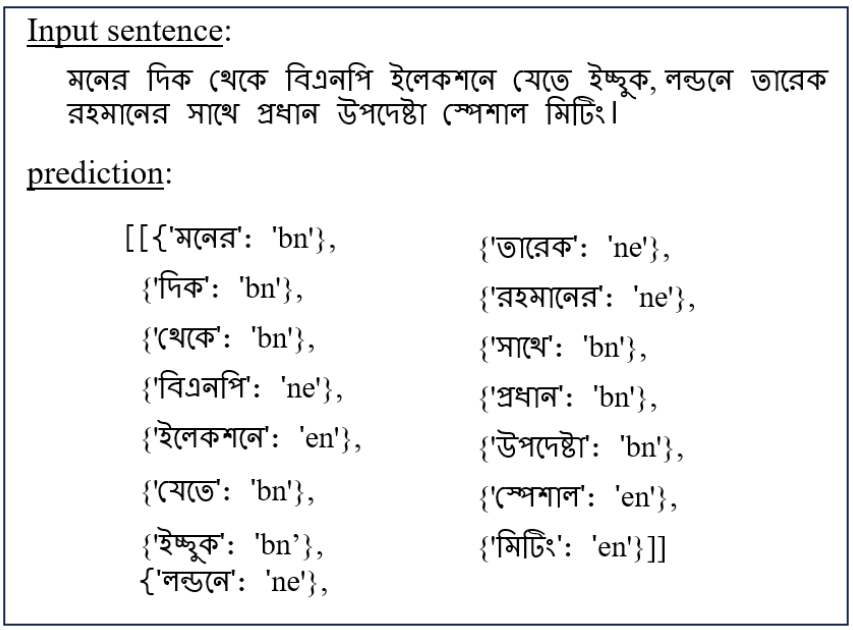

Figure 4.10: Prediction of different tags from input sentence

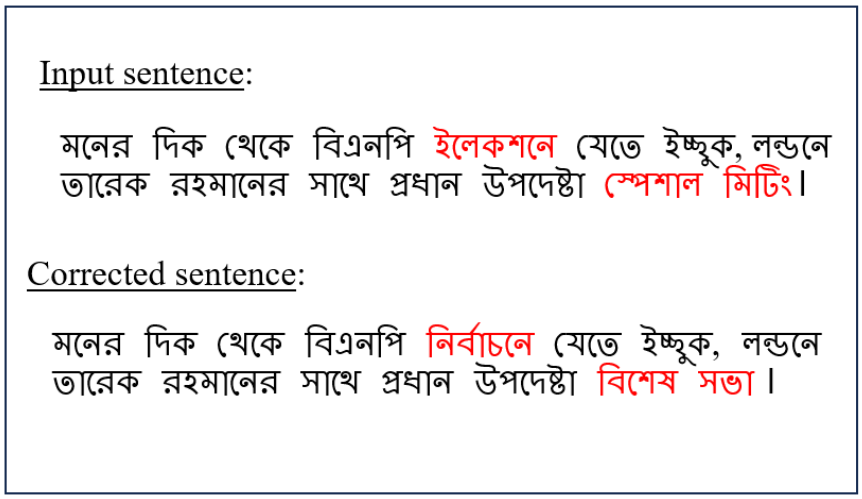

Figure 4.11: Conversion: English to standard Bangla

4.5 Summary

A modified Bert-base-NER model was utilized to recognize Bangla and English words from Bangla texts and convert English words into their corresponding standard Bangla words using Google Translator API. The system's performance was measured using criteria such as precision, accuracy, recall, and the F1 score. We also applied the input dataset to the prevailing machine learning (SVM, Naive Bayes) and deep learning (CNN+LSTM, BiLSTM) models to evaluate the proposed method. Our experiments (from Tables 4.7 and 4.11) demonstrate that, particularly in mixed-language situations, conventional models such as Naive Bayes have limitations when dealing with extremely contextual and morphologically complex languages like Bengali. Because deep learning models like CNN+LSTM and BiLSTM can capture contextual and sequential data, they produced higher outcomes. But the BERT-based model, which used its self-attention mechanism and pre-trained contextual embeddings to better capture the subtleties of code-mixed language, fared noticeably better than them.

The suggested Bert-base-NER model outperformed the baseline model from Borhan et al. [1], with precision of 94%, recall of 94%, F1-Score of 94%, and accuracy of 95%. This illustrates how well transformer-based architectures perform complicated natural language processing (NLP) tasks with multilingual or code-mixed datasets. In conclusion, these findings demonstrate that transformer designs fine-tuned on domain-specific datasets are a more reliable and effective solution for named entity recognition and language identification, particularly in low-resource and code-mixed language situations.

Chapter 5

Conclusion and Future Works

5.1 Summary

The objective of this research was to recognize Bangla and English words from bangla-english mixed Bangla texts and convert the recognized english words into standard bangla words using the Google Translator API. BN (Bangla), EN (English), NE (Named Entity), and UN (Unknown) were the four language tags that were used. In terms of precision, recall, and F1-score, the Bert-base-NER model outperformed the other models that were tested, including CNN+LSTM, BiLSTM, and Naive Bayes.

5.2 Conclusion

While linguistic evolution naturally involves adopting terms from other languages, excessive English-Bangla language mixing could compromise the language's uniqueness. It is important to use and promote the native language, particularly in literature, media, and education, in order to maintain Bengali's beauty and richness. Research on automatic language identification in code-mixed texts is very promising, particularly for low-resource language data. Recognition of English words in Bangla texts is important and the recognized English words converted into standard Bangla words will make the texts into standard Bangla texts. This research validates the effectiveness of transformer-based models, particularly Bert-base-NER, for recognition of Bangla and English words from Bangla-English mixed Bangla texts. The model successfully recognizes Bangla words, English words, named entity, and unknown words with high

accuracy. The performance improvement over baseline models underscores the importance of using context-aware architectures for low-resource languages like Bengali. This work not only contributes a strong baseline for Bangla and English word identification but also opens doors for future research in multilingual NLP applications, such as code-mixed sentiment analysis, machine translation, and speech recognition.

5.3 Limitations

Some limitations of proposed system:

- The size of the dataset is quite small.
- Transformer architecture makes it computationally costly.
- Different social media platforms may have different performance.
- Speech and multimodal data are not yet supported.

5.4 Applications

Our proposed system can be implemented in the following sectors for the standard Bengali language.

- Preprocessing for NLP Tasks in Code-Mixed Texts
- Education Sector
- Media
- Government sectors

5.5 Future Work

In our study, there is some room for improvement. We aim to do the following tasks to improve our system and for more accurate results:

- Increase the dataset size with more diverse domains and user-generated texts.

- Fine-tuning some multilingual models like mBERT or XLM-RoBERTa for even broader applicability.
- We aim to implement a lemmatization for the conversion of English to standard bangla words to make our system more accurate and reliable.

Bibliography

[1] B. Uddin, M. S. Hasan, M. S. Mia, M. A. Rahaman, M. P. Hossain, and F. Al Faisal, "A robust approach to identify banglish words using bangla scripts," in *2022 4th International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2022, pp. 1–6.

[2] R. P. Kumar, R. Elakkiya, R. Venkatakrishnan, H. Shankar, Y. S. Harshitha, K. Harini, M. N. Reddy *et al.*, "Transformer-based models for language identification: A comparative study," in *2023 International Conference on System, Computation, Automation and Networking (ICSCAN)*, 2023, pp. 1–6.

[3] A. L. Tonja, M. G. Yigezu, O. Kolesnikova, M. S. Tash, G. Sidorov, and A. Gelbukh, "Transformer-based model for word level language identification in code-mixed kannada-english texts," in *Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts*, 2022, pp. 18–24.

[4] B. Fetahu, A. Fang, O. Rokhlenko, and S. Malmasi, "Gazetteer enhanced named entity recognition for code-mixed web queries," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1677–1681.

[5] R. Priyadharshini, B. R. Chakravarthi, M. Vegupatti, and J. P. McCrae, "Named entity recognition for code-mixed indian corpus using meta embedding," in *2020 6th international conference on advanced computing and communication systems (ICACCS)*, 2020, pp. 68–72.

[6] C. Sabty, M. Elmahdy, and S. Abdennadher, "Named entity recognition on arabic-english code-mixed data," in *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, 2019, pp. 93–97.

[7] K. Shanmugavadivel, S. H. Sampath, P. Nandhakumar, P. Mahalingam, M. Subramanian, P. K. Kumaresan, and R. Priyadharshini, "An analysis of machine learning models for sentiment analysis of tamil code-mixed data," *Computer Speech & Language*, vol. 76, p. 101407, 2022.

[8] N. H. Mahadzir, M. F. Omar, M. N. M. Nawi, A. A. Salameh, and K. C. Hussin, "Sentiment analysis of code-mixed text: a review," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 3, pp. 2469–2478, 2021.

[9] N. Sabri, A. Edalat, and B. Bahrak, "Sentiment analysis of persian-english code-mixed texts," in *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, 2021, pp. 1–4.

[10] B. Bharathi *et al.*, "Ssncse_nlp@ dravidianlangtech-eacl2021: Offensive language identification on multilingual code mixing text," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, 2021, pp. 313–318.

[11] F. Balouchzahi, B. Aparna, and H. Shashirekha, "Mucs@ dravidianlangtech-eacl2021: Cooli-code-mixing offensive language identification," in *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, 2021, pp. 323–329.

[12] M. Kowsher, A. A. Sami, N. J. Prottasha, M. S. Arefin, P. K. Dhar, and T. Koshiba, "Bangla-bert: transformer-based efficient model for transfer learning and language understanding," *IEEE Access*, vol. 10, pp. 91 855–91 870, 2022.

[13] S. Sarker, "Banglabert: Bengali mask language model for bengali language understanding," 2020. [Online]. Available: https://github.com/sagorbrur/bangla-bert

[14] A. F. Hidayatullah, A. Qazi, D. T. C. Lai, and R. A. Apong, "A systematic review on language identification of code-mixed text: techniques, data availability, challenges, and framework development," *IEEE access*, vol. 10, pp. 122 812–122 831, 2022.

[15] F. Balouchzahi, S. Butt, A. Hegde, N. Ashraf, H. Shashirekha, G. Sidorov, and A. Gelbukh, "Overview of coli-kanglish: Word level language identification in code-mixed kannada-english texts at icon 2022," in *Proceedings of the 19th International Conference on Natural Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts*, 2022, pp. 38–45.

[16] S. Thara and P. Poornachandran, "Transformer based language identification for malayalam-english code-mixed text," *IEEE Access*, vol. 9, pp. 118 837–118 850, 2021.

[17] S. Gundapu and R. Mamidi, "Word level language identification in english telugu code mixed data," *arXiv preprint arXiv:2010.04482*, 2020.

[18] I. Chaitanya, I. Madapakula, S. K. Gupta, and S. Thara, "Word level language identification in code-mixed data using word embedding methods for indian languages," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2018, pp. 1137–1141.

[19] S. D. Das, S. Mandal, and D. Das, "Language identification of bengali-english code-mixed data using character & phonetic based lstm models," in *Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation*, 2019, pp. 60–64.

[20] N. Sarma, S. R. Singh, and D. Goswami, "Word level language identification in assamese-bengali-hindi-english code-mixed social media text," in *2018 International Conference on Asian Language Processing (IALP)*, 2018, pp. 261–266.

[21] T. Asubiaro, T. Adegbola, R. Mercer, and I. Ajiferuke, "A word-level language identification strategy for resource-scarce languages," *Proceedings of the Association for Information Science and Technology*, vol. 55, no. 1, pp. 19–28, 2018.

[22] D. Nguyen and A. S. Dogruöz, "Word level language identification in online multilingual communication," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, 2013, pp. 857–862.

[23] H. Jhamtani, S. K. Bhogi, and V. Raychoudhury, "Word-level language identification in bi-lingual code-switched texts," in *Proceedings of the 28th Pacific Asia Conference on language, information and computing*, 2014, pp. 348–357.

[24] A. Dutta, "Word-level language identification using subword embeddings for code-mixed bangla-english social media data," in *Proceedings of the Workshop on Dataset Creation for Lower-Resourced Languages within the 13th Language Resources and Evaluation Conference*, 2022, pp. 76–82.

[25] K. Shanmugalingam, S. Sumathipala, and C. Premachandra, "Word level language identification of code mixing text in social media using nlp," in *2018 3rd international conference on information technology research (ICITR)*, 2018, pp. 1–5.

[26] A. K. Nandanwar and J. Choudhary, "Contextual embeddings-based web page categorization using the fine-tune bert model," *Symmetry*, vol. 15, no. 2, p. 395, 2023.

[27] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. Ali Almazroi, "A fine-tuned bert-based transfer learning approach for text classification," *Journal of healthcare engineering*, vol. 2022, no. 1, p. 3498123, 2022.