Shahjalal University of Science and Technology
Department of Computer Science and Engineering

A Comprehensive Approach for English(Bangla phonetic)-to-Bangla Transliteration Using Deep Learning

Md Mizbah Uddin Junaed

Reg. No.: 2018331115

4^{th} year, 2^{nd} Semester

Marajul Islam Shawn

Reg. No.: 2018331080

4^{th} year, 2^{nd} Semester

Department of Computer Science and Engineering

Supervisor

Mohammad Abdullah Al Mumin, PhD

Professor

Department of Computer Science and Engineering

25th February, 2024

A Comprehensive Approach for English(Bangla phonetic)-to-Bangla Transliteration Using Deep Learning

A Thesis submitted to the Department of Computer Science and Engineering, Shahjalal University of Science and Technology, in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

By

Md Mizbah Uddin Junaed
Reg. No.: 2018331115
4^{th} year, 2^{nd} Semester

Marajul Islam Shawn
Reg. No.: 2018331080
4^{th} year, 2^{nd} Semester

Department of Computer Science and Engineering

Supervisor

Mohammad Abdullah Al Mumin, PhD
Professor
Department of Computer Science and Engineering

25th February, 2024

Recommendation Letter from Thesis/Project Supervisor

The thesis/project entitled *Thesis/Project title* submitted by the students

1. Md Mizbah Uddin Junaed
2. Marajul Islam Shawn

is under my supervision. I, hereby, agree that the thesis/project can be submitted for examination.

Signature of the Supervisor:

Name of the Supervisor: Mohammad Abdullah Al Mumin, PhD

Date: 25th February, 2024

Certificate of Acceptance of the Thesis/Project

The thesis/project entitled *A Cross-Linguistic and Comprehensive Approach for English-to-Bangla Phonetic Transliteration* submitted by the students

1. Md Mizbah Uddin Junaed
2. Marajul Islam Shawn

Head of the Dept.	Chairman, Exam. Committee	Supervisor
Md Masum	Md Masum	Mohammad Abdullah Al
Professor and Head	Professor and Head	Mumin, PhD
Department of Computer	Department of Computer	Professor
Science and Engineering	Science and Engineering	Department of Computer
		Science and Engineering

Abstract

Machine transliteration plays a vital role in cross-language applications, particularly in languages with distinct phonetic representations like English to Bengali. This study addresses the challenges and significance of cross-linguistic communication, focusing on English-to-Bangla transliteration. In the digital era, language adaptation dynamics underscore the need for accurate transliteration to preserve linguistic integrity and cultural identity. Our research centers on a transliteration process considering all aspects of Bengali phonetic script in the English alphabet, widely used by Bengali speakers. The aim is to convert English script (Bengali phonetic) into the original Bengali script using efficient techniques and a deep learning approach. Meticulous data collection, cleaning, tokenization, and subword segmentation form the research methodology, creating a diverse dataset of Bangla phonetic-to-Bangla transliteration pairs. Precise alignment ensures transliteration consistency. Statistical models and deep learning capture phonetic nuances and contextual variations. A comparative analysis of previous studies revealed a gap in transliteration processes lacking precision and comprehensive consideration of spelling aspects. By harnessing neural networks and NLP, our approach offers a solution to intricate cross-linguistic transliteration challenges, bridging linguistic and cultural gaps. The developed system aims to foster precise, culturally sensitive communication as languages evolve in the digital age.

Keywords: MT, cross-language applications, linguistic integrity, phonetic representations, deep learning techniques, phonetic nuances, contextual variations, linguistic diversity, mt5, flax-t5, banglat5.

Acknowledgements

At first we express our sincere appreciation to Mohammad Abdullah Al Mumin, PhD, Professor of CSE at SUST, for his invaluable guidance and mentorship throughout the semester.
We extend our gratitude to the Department of Computer Science and Engineering at Shahjalal University of Science and Technology, located in Sylhet 3114, Bangladesh, for their valuable support in this thesis.It is important to acknowledge the contributions of previous researchers in this field, as their works has greatly influenced our own research endeavors. We also would like to extend our appreciation to the numerous online resources that have been instrumental in this research work. We acknowledge the valuable insights, data, and references obtained from various online databases, research articles, academic journals, and reputable websites. We would especially like to mention our seniors Mahdi Hasan for some guidance.
Afterall, we are thankful to the Almighty for His blessings, which made this work possible.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Bangla, or Bengali, spoken by approximately 210 million people predominantly in Bangladesh and the Indian state of West Bengal, ranks as the fourth most widely spoken language globally. As a member of the Indo-European family, specifically the Aryan or Indo-Iranian branch, Bangla has evolved over time, resulting in a notable disparity between its script and pronunciation.

The pervasive influence of online communication, facilitated by social platforms, has become integral to daily life, fostering unity and shaping the expression of thoughts and emotions in virtual spaces.

The transliteration of English (Bangla phonetic) script into the Bangla script plays a crucial role in various natural language processing (NLP) applications, including machine translation and information retrieval. This task is particularly challenging when translating names and technical terms between languages with distinct alphabets and sound inventories. To address this challenge, entities are often transliterated to approximate their phonetic equivalents. The phonemic inventory of Bangla presents irregularities and complexities, characterized by diverse nature and consonant clusters.

In this study, our objective is to establish a comprehensive transliteration framework, incorporating Direct phonetic mapping and Hybrid transliteration modeling with a Markov assumption. This framework aims to enhance the accuracy and efficiency of transliteration and back transliteration processes, addressing the intricate phonetic variations inherent in the Bangla language.

1.2 Motivation

The surge of Banglish as a dominant online communication mode prompts the exploration of its impact on linguistic integrity and cultural continuity. The motivations behind this research are deeply rooted in the preservation of linguistic heritage and the maintenance of cultural identity:

Look at the scenarios :

Understanding native bangla phonetic text (eg. Ame, ami)

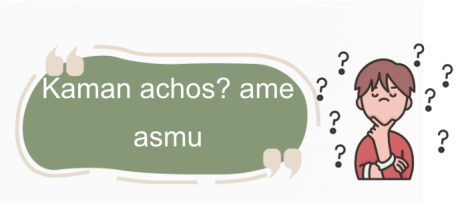


Figure 1.1: A scenario

You want to say something in bangla but no input method!

Figure 1.2: Another scenario

- **Linguistic Diversity and Adaptation :** The emergence of Banglish underscores the adaptability of languages in response to changing communication dynamics. The linguistic blend reflects the creativity and resilience of language users in navigating digital spaces, reshaping linguistic norms, and creating new forms of expression.
- **Cultural Reflection and Representation :** Language is a reflection of culture and identity. Banglish serves as a reflection of the cultural interplay in Bangladesh, where diverse linguistic backgrounds converge. Studying Banglish-to-Bangla transliteration is a step toward accurately representing this linguistic and cultural complexity.
- **Linguistic Integrity :** As Banglish becomes a mainstay in online communication, ensuring the integrity of its linguistic components is crucial. Developing a robust machine transliter-

ation system safeguards against the erosion of linguistic nuances, maintaining the phonetic and semantic essence of both languages.

- **Preserving Cultural Heritage :** The Bangla script is not merely a means of communication; it carries the historical, social, and cultural legacy of Bangladesh. Effective Banglish-to-Bangla transliteration contributes to the preservation of this heritage, allowing users to communicate in a script that resonates with their cultural roots.

1.3 Research Objectives

This research endeavors to address the following core objectives:

- **Transliteration System Development:** To design and implement an accurate and efficient machine transliteration system that seamlessly converts Banglish text into the native Bangla script.
- **Linguistic Precision through NLP:** To harness the capabilities of Natural Language Processing (NLP) techniques to enhance the transliteration system's accuracy, ensuring that the intricacies of both languages are captured faithfully.
- **Validation and Significance:** To validate the developed system's performance using real-world Banglish data and to underline the significance of accurate transliteration for preserving linguistic integrity and cultural continuity.

1.4 Bangla Phonetic Transliteration

There exist various forms of transliteration. Our research focuses on the **Backward Transliteration** that involves converting a loanword from a foreign language back to its original language's phonetic representation. Conversely forward Transliteration, which pertains to the conversion of a word from its native language to a foreign language through a phonetic representation.
The classification of transliteration approaches can be divided into three types:
grapheme-based, phoneme-based, and hybrid models, as well as correspondence-based transliteration models.

These models are categorized based on the units that are being transliterated. In grapheme-based transliteration models, the mapping is done directly from source graphemes to target graphemes, without considering the phonetic aspects of the source language words. Various methods have been proposed based on this model, such as source-channel models, decision trees, transliteration networks, and joint source-channel models. On the other hand, phoneme-based transliteration models focus on the pronunciation or source phonemes rather than the spelling or source graphemes. This model involves transforming source graphemes to source phonemes and then to target graphemes. Approaches like weighted Finite State transducers (WFST) and extended Markov window fall under the phoneme-based models. These models consider transliteration as a phonetic process rather than an orthographic one, requiring two steps: generating source language phonemes from source language graphemes and producing target language graphemes from source phonemes. Hybrid transliteration models and correspondence-based transliteration models utilize both source graphemes and source phonemes in machine transliteration. The correspondence-based model utilizes the correspondence between a source grapheme and a source phoneme when generating target language graphemes, while the hybrid models combine grapheme information and phoneme information through linear interpolation. It is important to note that the hybrid model combines the probabilities of grapheme-based transliteration and phoneme-based transliteration using linear interpolation.

Chapter 2

Background Study

2.1 What is Transliteration?

Transliteration is the process of converting text from one writing system to another while maintaining the phonetic or sound-based correspondence between the characters of the two systems. It involves representing the same spoken language using the characters of a different script or alphabet.

Unlike translation, which focuses on conveying the meaning of words or sentences, transliteration is concerned with representing the sounds of words accurately in a different script. This can be particularly useful when dealing with proper nouns, names, technical terms, or specialized vocabulary that may not have direct equivalents in another language.

The transliteration from English letters is of particular importance for users who are only familiar with the English keyboard layout. These users may struggle to type quickly in a different alphabet, even if their software supports a keyboard layout for another language. This is the primary purpose of transliteration. In the case of Bangla, also known as Bengali, there are various keyboard layouts, making it difficult for beginners to memorise and write smoothly. While there are some phonetic keyboard layouts available to assist beginners, a reliable transliteration process is still necessary. Bangla is not a highly phonetic language, meaning that its orthographic rules do not always align with its phonetic rules. This discrepancy can lead to silent letters, letters taking on the sound of another letter, and letters sounding differently depending on the context. These complexities make it challenging to obtain the correct dictionary word with the intended pronunciation, even when

writing in English. In this section, we will discuss how we can achieve accurate transliterations of complex dictionary words and regular words based on their English pronunciation.

Transliteration vs Transcription

In a strict sense, transliteration refers to the process of converting one script into another script while aiming to maintain the original spelling of unknown transliterated words. This is in contrast to transcription, which involves mapping the sounds of one language to the script of another language. However, it is common for transliterations to map the letters of the source script to letters that are pronounced similarly in the target script, specifically for a particular pair of source and target languages. In a more specialized context, transcription involves writing the sounds of a word in one language using the script of another language. If the relationship between letters and sounds is similar in both languages, a transliteration may be nearly identical to a transcription. In a broader sense, the term transliteration encompasses both the narrow sense of transliteration and transcription.

Example of transliteration

Here is an example of bengali text written in english alphabet taken from social media that is to be transliterated into its corresponding original bengali script: (table)

Bangla in English Alphabet	Corresponding Bengali script
ajke sob kicho eto nirob kno?	আজকে সব কিছু এত নিরব নিরব কেন?
sb kichotei pansha ekta shad onuvob hoitese	সব কিছুতেই পানসা পানসা একটা স্বাদ অনুভব হইতেছে
amar ekar hoitese naki apnadero?	আমার একার হইতেছে নাকি আপনাদেরও?

Figure 2.1: Example of transliteration

Now converting these text into their **Corrensponding Bengali script** can be considered to be an example of transliteration.

Transliterating a word from the language of its origin to a foreign language is called Forward Transliteration, while transliterating a loan-word written in a foreign lan- guage back to the language

of its origin is called Backward Transliteration. [1]

Transliteration approaches can be categorized into three main types: Grapheme-based, Phoneme-based, and Hybrid models, including Correspondence-based transliteration models.

Grapheme-based Transliteration : Direct orthographical mapping from source graphemes to target graphemes without relying on phonetic knowledge.
Methods: Utilizes a direct approach, transforming source language graphemes into target language graphemes. Examples include source-channel models, decision trees, transliteration networks, and joint source-channel models. Characteristic: Does not involve phonetic information of source language words.

Phoneme-based Transliteration : Focuses on pronunciation or source phonemes rather than spelling or source graphemes.
Methods: Involves source grapheme-to-source phoneme and source phoneme-to-target grapheme transformations. Techniques like Weighted Finite State Transducers (WFST) and extended Markov window fall under this category.
Characteristic: Treats transliteration as a phonetic process, requiring two steps: generating source language phonemes from source language graphemes and producing target language graphemes from source phonemes.

Hybrid Transliteration and Correspondence-based Transliteration : Combine source graphemes and source phonemes in machine transliteration.
Hybrid Model: Integrates grapheme-based and phoneme-based transliteration probabilities using linear interpolation. Correspondence-based Model: Leverages the correspondence between a source grapheme and a source phoneme during the generation of target language graphemes.
Characteristic: Hybrid models blend grapheme and phoneme information, while correspondence-based models use the relationship between source graphemes and source phonemes in the transliteration process.
In summary, these transliteration models offer diverse approaches, catering to different aspects of the source language, be it orthographical or phonetic, and they demonstrate varied techniques for achieving accurate transliteration.

2.2 Transliteration Techniques for Cross-Linguistic Communication

The intricate relationship between language, script, and phonetics underpins the field of transliteration, which enables the conversion of text from one script to another while preserving phonetic attributes. The proposed thesis focuses on the transliteration of Bangla, an Indo-Aryan language, from the English alphabet to the original Bangla script. This task is particularly challenging due to the distinct phonological features and character combinations of both languages.

Linguistic Foundations:

The Bangla script, an abugida system, combines consonants and vowels to form syllabic units. Unlike alphabetic scripts, Bangla characters often represent complex phonemes, demanding a nuanced approach to transliteration. To overcome these challenges, the thesis employs a phonetic mapping strategy that encapsulates the phonetic variations of Bangla characters as they are transliterated into the English alphabet.

Cross-Linguistic Transliteration Challenges:

The diversity in phonetics and orthography between English and Bangla presents formidable hurdles in achieving accurate transliteration. Traditional character-based methods fall short in capturing phonetic nuances and contextual variations. The neural network-based approach in this thesis takes a holistic view, leveraging deep learning techniques to discern intricate phonetic patterns and relationships, thereby producing more precise transliteration results.

Encoding Techniques:

The neural network's effectiveness hinges on encoding techniques that convert textual input into numerical representations amenable to machine learning algorithms. In the proposed thesis work, encoding plays a crucial role in facilitating accurate transliteration. By embedding phonetic and contextual information, the encoding process enables the neural network to grasp the phonetic subtleties of both languages, enhancing the transliteration quality.

Comprehensive Approach:

At the heart of the proposed approach lies its comprehensiveness. Unlike conventional methods that focus on individual character replacement, this approach considers the broader phonetic context. By analysing neighbouring characters and their phonetic interactions, the neural network identifies complex phonetic mappings, enabling accurate and contextually-aware transliteration. This innovation aligns with the thesis's aim of achieving precise and linguistically informed cross-

linguistic transliteration.

The proposed neural network-based method, empowered by phonetic mapping and encoding, seeks to revolutionise the field of transliteration, bridging the gap between languages and scripts and fostering more accurate cross-linguistic communication.

2.2.1 Transformer Model

The transformer model is a neural network architecture introduced in the paper "Attention Is All You Need" by Vaswani et al. (2017). It has revolutionized various natural language processing tasks, including machine translation, due to its ability to capture long-range dependencies and contextual relationships in sequences efficiently. The architechture is shown in the figure **??** ref: https://www.researchgate.net/figure/Transformer-Model-Architecture-Transformer-Architecture [accessed 23 Feb, 2024]

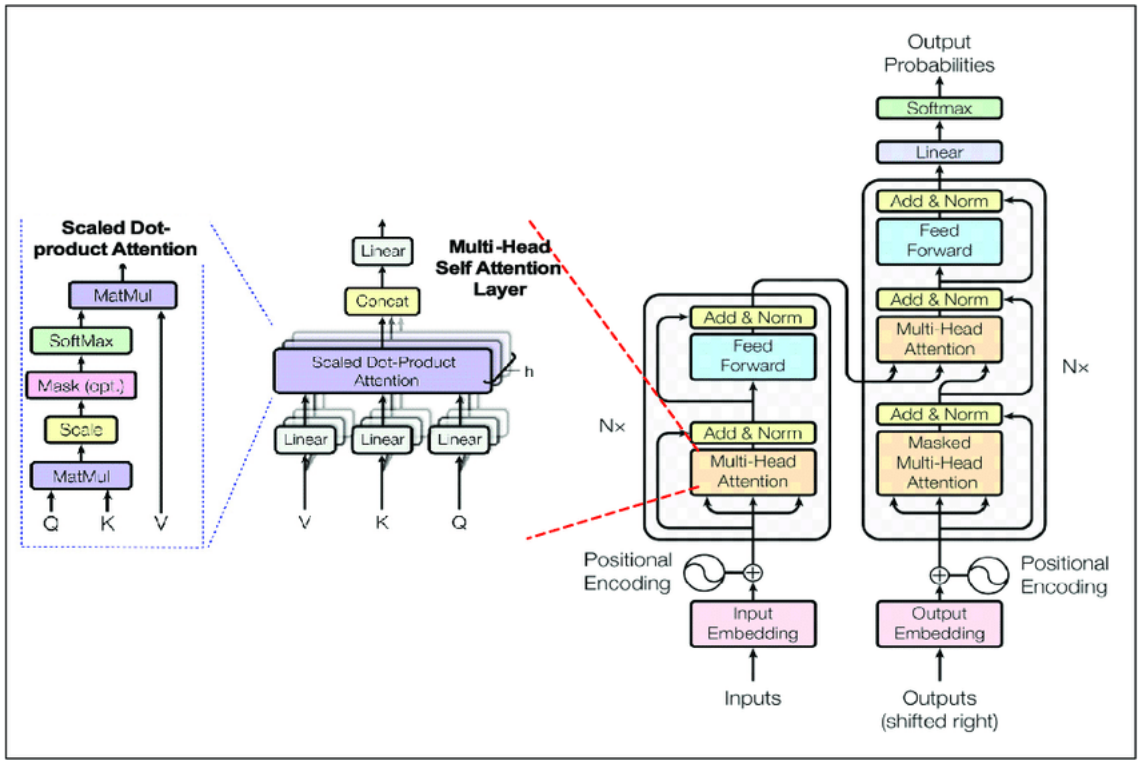


Figure 2.2: Architechture of Transformer model

Key Components:

- **Self-Attention Mechanism:** At the heart of the transformer architecture is the self-attention

mechanism, which allows the model to weigh the importance of different words in a sequence relative to each other. It computes attention scores for each word in the sequence based on its relationships with other words.

- **Encoder and Decoder:** The transformer model is commonly divided into an encoder and a decoder. The encoder processes the input sequence and encodes its information into a series of context-rich representations. The decoder takes these representations and generates the output sequence, step by step.

- **Multi-Head Attention:** To capture different types of relationships and dependencies, the self-attention mechanism is applied multiple times in parallel, each time with different learned linear projections.

- **Positional Encoding:** Since the transformer model lacks any inherent notion of sequence order, positional encodings are added to the word embeddings to provide information about their positions in the sequence. This enables the model to learn the sequential relationships between words.

- **Feed-Forward Neural Networks:** After attention layers, the model employs feed-forward neural networks to process the context-rich representations further. These networks include non-linear transformations that help the model capture complex patterns in the data.

- **Residual Connections and Layer Normalization:** Each sub-layer (self-attention, feed-forward network) in the transformer architecture has residual connections around it, followed by layer normalization. These mechanisms stabilize training and enable the model to learn more effectively.

2.2.2 t5 Model

It is a Transformer-based model that uses a text-to-text approach. This means that the model is trained to convert text from one language to another.
The T5 model has been shown to be effective on a variety of machine translation tasks. It has been shown to achieve state-of-the-art results on the WMT benchmark, which is a standard benchmark for machine translation. The architechture is shown in the figure **??** ref:

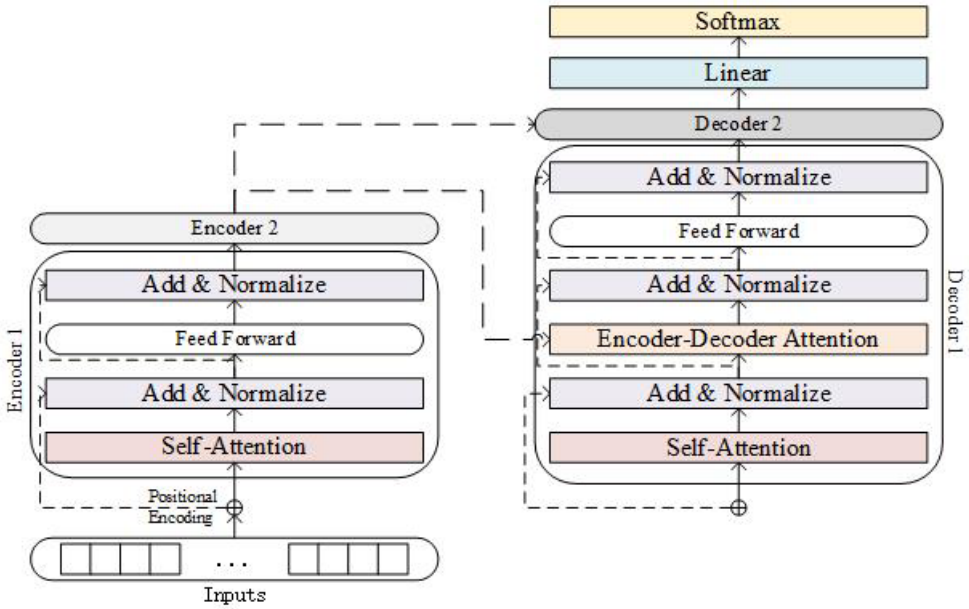


Figure 2.3: Architechture of t5 model

https://www.researchgate.net/figure/t5 [accessed 23 Feb, 2024]

However, T5 model also has some limitations for machine translation. One limitation is that it can be difficult to fine-tune the model for specific language pairs. This is because the model is trained on a massive dataset of text and code, which includes many different languages. Another limitation of the T5 model is that it can sometimes generate outputs that are not grammatically correct or idiomatic. This is because the model is not explicitly trained on grammatical rules or idiomatic expressions.

Despite these limitations, the T5 model is a powerful tool that can be used for machine translation. It is still under development, but it has already shown great promise. Here are some other limitations of the T5 model for machine translation:

- The model can be computationally expensive to train and fine-tune.
- The model can be sensitive to the quality of the training data.
- The model can be biased, reflecting the biases in the training data.

Chapter 3

Related Works

In this research, we analysed about 26 related research works. Most of them are on RMT and SMT. Some studies used NMT but did not handle all aspects, eg. only works with name entities.

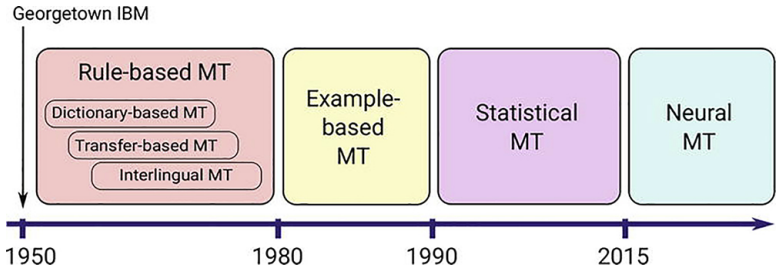


Figure 3.1: MT evolution

There is a work on transliteration where the proposed methodology [2] utilizes an alignment representation to address the transliteration challenge in a low-resource language pair. The approach involves three primary steps: (1) pre-processing, (2) modifying the input sequences using alignment representation, and (3) developing a machine transliteration system based on recurrent neural networks (RNNs) for enhanced efficiency.

This study [3] explores Bengali to English neural machine translation using LSTM, GRU, BiLSTM, and BiGRU models, finding that BiLSTM yields the best results. With BLEU scores of 47.4, 35.8, 32.0, and 22.8 for BLEU-1, 2, 3, and 4 respectively, the study demonstrates notable advancements in translation accuracy. This study explores Bengali to English neural machine translation using LSTM, GRU, BiLSTM, and BiGRU models, finding that BiLSTM yields the best

results. With BLEU scores of 47.4, 35.8, 32.0, and 22.8 for BLEU-1, 2, 3, and 4 respectively, the study demonstrates notable advancements in translation accuracy.

This paper [4] introduces enhancements to an English to Bengali statistical machine translation system, including transliteration and preposition handling modules, which improve translation accuracy. While the transliteration module achieves moderate accuracy and enhances the impression of translation quality, the preposition handling module also contributes to accuracy improvements. However, overall scores remain relatively low, with BLEU averaging at 11.70, NIST at 4.27, and TER at 0.76.

The paper [5] uses transliteration for a simple and novel approach for steganography. The key concept of this approach is to utilize the unique characteristic of Bengali phonetic keyboard layouts to conceal confidential data as bits. By assigning one option to represent '0' and another option to represent '1', secret information can be hidden in a document without being comprehensible to any unauthorized user.

The work [6] implemented phonetic encoding for Bangla, taking into account the various context-sensitive rules, including those involving the large repertoire of conjuncts in Bangla. Two methods are applied: Direct mapping, Phonetic mapping.

In this article [7], the authors introduce a technique for the automated acquisition of a transliteration model using a dataset of name pairs from two distinct languages. They proceed to assess the effectiveness of the model and conduct a comparative analysis of three variations for English to Arabic transliteration.

The applied statistical transliteration techniques, specifically the Monogram Transliteration Model and the Bigram Transliteration Model.

In this study [8], they presented a direct approach for transliterating English to Chinese. Two direct transliteration models are suggested. The first model treats the problem as a direct mapping from English phonemes to basic Chinese phonetic symbols, with additional mapping units discovered during the training process. An algorithm for aligning phoneme chunks is introduced. The second model incorporates contextual features of each phoneme using Maximum Entropy formalism, and is further improved with a precise alignment scheme based on phoneme chunks. The direct approaches are compared to a source-channel baseline using the IBM SMT model, and

it is demonstrated that the second approach is significantly better.
This work [9] also introduced a new estimation algorithm and showcase its exceptional precision across diverse databases. Additionally, they investigate the influence of the maximum approximation in training and transcription, the interplay of model size parameters, the generation of n-best lists, confidence measures, and the conversion of phonemes to graphemes.
Another work [10] purposes to develop a system that is much better to check the spelling of the transliterated word by calculating Levenshtein distance. To make the mechanism more efficient and accurate unigram method has been implemented.

There is recent work [11] that introduces a unique pipeline that uses nine open source back transliteration tools to automatically back transliterate Romanized Bengali to Bengali.âThe pipeline consists of seven steps: (1) processing the Romanized Bengali input; (2) acquiring human transliteration for performance comparison; (3) employing transliteration tools; (4) generating candidatetransliterations; (5) post-processing the candidate transliterations; (6) selecting best candidate transliteration, and (7) evaluating the quality of the transliterations through several performance metrics.

This [12] explores the possibility of using multilingual pretrained transformers like mBART and mT5 for exploring one such task of code-mixed Hinglish to English machine translation. Further, we compare our approach with the only baseline over the PHINC dataset and report a significant jump from 15.3 to 29.5 in BLEU scores, a 92.8% improvement over the same dataset.

3.1 Different transliteration tools

Bengali Phonetic Parser :
Utilizes a simple phonetic level implementation to generate phonetic spellings for Bengali words, facilitating transliteration.
Drawback: Limited in its complexity and may not handle more nuanced transliteration cases effectively.

pyAvroPhonetic : Adapts the popular Bengali phonetic-typing software Avro Keyboard to automatically transliterate Romanized Bengali into Bengali.
Drawback: Relies on the functionality of Avro Keyboard, which may not cover all transliteration

scenarios comprehensively.

Google Translate : Employs a large neural network model, Google Neural Machine Translation (GNMT), to support transliteration among 133 languages. Drawback: Limited control over transliteration process and potential inaccuracies in handling specific transliteration cases.

Indic Transliteration : Offers transliteration functions for Sanskrit to convert text from Latin script to various Indian scripts, including Bengali.

Drawback: Complex tool with multiple romanization styles may require expertise for effective utilization.

BNTRANSLIT: Utilizes deep learning with an attention-based LSTM architecture to transliterate Romanized Bengali words into Bengali.

Drawback: Despite training on a large dataset, accuracy may still be limited in capturing all transliteration variations accurately.

Google Transliteration IME: Provides a virtual keyboard for direct typing in native language text, using dictionary-based phonetic transliteration.

Drawback: Reliance on dictionary-based transliteration may result in inaccuracies for uncommon or context-specific transliteration cases.

GPT-3: Employs transformer-based neural network models pre-trained on a vast corpus of text data to perform transliteration tasks.

Drawback: Requires payment for usage beyond certain limits, limiting accessibility for widespread use. Additionally, its performance may vary based on the amount of training data provided.

3.2 Overview of Previous Research

Machine transliteration has been extensively researched for various language pairs. Different approaches have been developed based on the characteristics of the languages involved. The development of algorithms for machine transliteration started more than ten years ago, focusing on the phonetics of the source and target languages. This was followed by statistical and language-specific methods.

There are several transliteration applications available for Bangla, however, almost all of them provide a one-to-one mapping. This means that each English letter or letters are converted to a fixed Bangla letter or letters. Unfortunately, there are no transliteration options currently available

that can generate dictionary words with the same pronunciation when written in English.

Previous research has focused on the transliteration of English words into other languages, presenting a significant research challenge. Numerous studies have been conducted in this field, including transliterations from English to Japanese [13], English to Arabic [7] [14], and English to Chinese [8]. These transliterations have found application in various domains, such as multilingual information retrieval and the identification of out-of-vocabulary words with similar pronunciation through statistical analysis. The initial efforts in transliterating Bangla were pioneered by ITRANS https://www.aczoom.com/itrans/online/ in the early 1990s. Presently, this application is gaining popularity and proving to be useful. However, these methods employ a phonetically direct mapping, which involves mapping from one script to another without any loss of information. To date, no research has been found on phonetic mapping, which would provide words with the same pronunciation from a dictionary.

In this paper [7], the authors introduce and assess a straightforward statistical method for English to Arabic transliteration. This method does not rely on any heuristics or linguistic expertise in either language. Instead, it learns the probabilities of character translations between English and Arabic from a training dataset consisting of pairs of transliterated words from both languages. Using these probabilities, the method generates Arabic transliterations for unfamiliar English words. The authors compare a context-dependent version of the model with a context-independent version. It is worth noting that these models primarily focus on proper nouns and may not encompass all aspects of words. Nonetheless, this technique has the potential to be applied to any language pair.

A english-chinese transliteration model [8] introduced Two direct transliteration models. The initial approach considers the problem as a direct conversion from English phonemes to basic Chinese phonetic symbols, with the possibility of discovering additional conversion units during training. It introduces an algorithm for aligning groups of phonemes. The second approach includes contextual characteristics of each phoneme using Maximum Entropy formalism and is enhanced with a precise alignment scheme based on phoneme groups. The direct methods are compared to a source-channel baseline using the IBM SMT model, and it is shown that the second approach is considerably superior.

Another system described in this paper [15] employs a fusion of two methodologies to perform direct grapheme-to-grapheme transliteration. This technique operates without any language-specific

assumptions, reliance on dictionaries, or explicit phonetic data. Instead, it directly converts sequences of tokens in the source language into corresponding sequences of tokens in the target language. But the transliteration method employed is based on phrases.

This paper [1] presents a survey on the advancements of various machine transliteration systems for Indian languages. Based on the survey conducted, we can discover that the majority of Indian language machine transliteration systems currently in existence rely on statistical and hybrid methodologies. The primary focus and difficulty in each development lies in designing a system that effectively incorporates the complex agglutinative and morphological characteristics of the language.

3.3 Comparative Analysis of Previous Works

Current methods for transliterating English into other languages, such as the IBM SMT model, are usually built on the source-channel framework. However, the accuracy of these models are relatively poor.

Due to the manipulation of individual letters and focus on syllabic pronunciation, the Metaphone and Double Metaphone algorithms are well-suited for detecting suggestions in Western languages with smaller and less complex alphabets. However, South Asian languages possess larger and more intricate alphabets. Consequently, in order to provide appropriate suggestions, it becomes necessary to simulate various representations of misspelled words based on sets of similarly spelled letters, vowel symbols, consonant symbols, and compound letters. Unfortunately, the Metaphone and Double Metaphone algorithms are not designed to meet these requirements. Therefore, they are not particularly suitable or efficient for South Asian and other languages. In light of these circumstances, there is a need for a new algorithm that can fulfill all the mentioned criteria and simplify the implementation of searching for the nearest suggestions for misspelled words in languages such as Bangla and other South Asian languages. As a partial solution to this problem, a new algorithm called RecursiveSimulation has been proposed.

RecursiveSimulation: Recursive simulation method is employed to group together Bangla letters that have similar sounds. This method takes into account both the letters themselves and the symbolic representation of vowels and consonants. This algorithm utilizes a set of circular lists, each containing phonetically similar letters. However, it is important to note that RecursiveSimulation

is still in the research and development phase.

The research [16] presents a transliteration system for person names written in Punjabi (Gurumukhi Script) that combines statistical and rule-based approaches. This system generates English (Roman Script) transliterations for Punjabi names. Experimental results have demonstrated a satisfactory level of performance, with an overall accuracy rate of 95.23%. However, it is important to note that the model primarily emphasizes rule-based transliteration.

This work [17] employed a modified joint source-channel model, as well as two other methods, to produce Hindi transliterations from English words, with the aim of generating a wider range of spelling variations for Hindi names. Additionally, a set of postprocessing rules were developed to eliminate errors. In the course of the experiment, a word accuracy of 0.471 and a mean F-score of 0.831 were achieved.

3.4 Identified Gaps in the Literature

Some works incorporate a one-to-one mapping approach that fails to encompass all facets of cross-linguistic communication.

Such as, a proposal [18] revolves around the implementation of a one-to-one character representation system. This system aims to facilitate the memorization and compilation of input text. While the system is specifically designed for Bangla text, the English characters have been carefully selected to closely correspond to the phonetics of Bangla characters.

This research paper [19] showcases the effectiveness of neural sequence-to-sequence models in achieving state-of-the-art or near state-of-the-art performance on established datasets. In order to enhance the accessibility of machine transliteration, we have made available a novel dataset for Arabic to English transliteration, along with the corresponding trained models, as open-source resources. Conducting a comparative analysis of epsilon insertion models and attentional sequence-to-sequence models across three benchmark datasets, the findings indicate that attentional sequence-to-sequence models generally exhibit superior performance, although not consistently across all cases [19].

Some research works [20], [21] presented several deep learning architectures utilizing neural networks for the transliteration of named entities. These architectures incorporate two distinct frameworks for neural machine translation (NMT): recurrent neural network and convolutional

sequence to sequence based NMT. The results demonstrate that this approach yields highly satisfactory outcomes in the realm of multilingual machine transliteration. The submitted runs consist of an ensemble of diverse transliteration systems for all language pairs. In the NEWS 2018 Shared Task on Transliteration, this method achieves superior performance for the En-Pe and Pe-En language pairs, while also delivering comparable results for other scenarios.

The [22] paper conducted the process of transliteration is through the utilization of an attention-based approach in deep learning. In contrast to prior studies that employed random weight initialization in the encoder, the initial values for the weights are derived from the word vector representation of the source vocabulary. This representation is obtained by calculating the co-occurrences between distinct characters. The experimental findings, based on an English to Persian transliteration corpus comprising over 14,000 word pairs, demonstrate the enhanced performance of the proposed method. Specifically, it achieves an improvement of up to 4.21 BLEU points compared to the basic attention-based approach.

Another interesting study [23] employs machine learning (ML) and deep learning (DL) models to classify transliterated Bengali comments on social media. To address the scarcity of publicly available transliterated Bengali data, we crafted a dataset of 1,300 comments, accessible on Mendeley Data. They assessed logistic regression (LR), SVM, decision tree, and neural networks. Among these, logistic regression with countVectorizer performed best, achieving an 85.76% accuracy and 85.70% F1 score. The use of a self-created dataset showcases the commitment to overcoming data limitations, and the results achieved by logistic regression underscore its efficacy in this context. Overall, this study contributes to the ongoing discourse on utilizing advanced techniques to analyze social media interactions.

Chapter 4

Dataset Collection and Preprocessing

In the field of thesis or research, the significance of datasets in the context of training a model cannot be overstated. Datasets serve as the base foundation upon which the success and reliability of a model reside. Diverse and real-world datasets ensure a well-trained model.

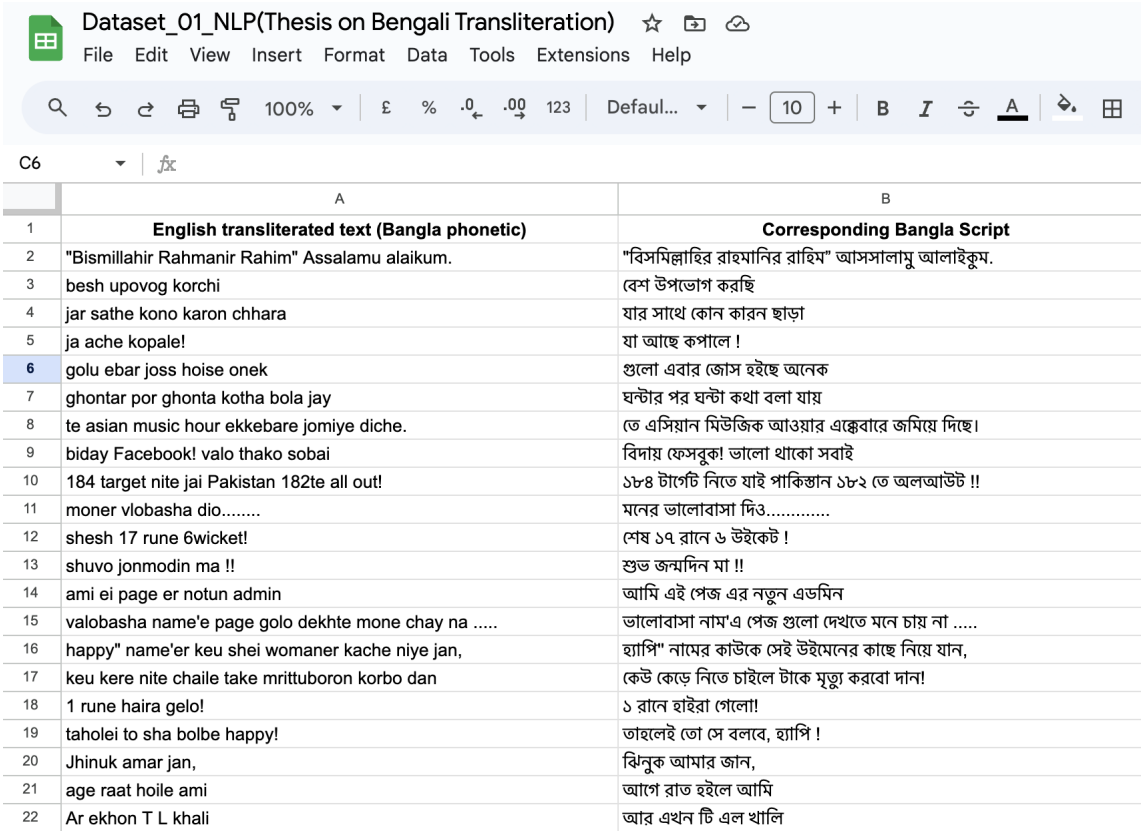
4.1 Data Collection

According to our findings, we manually collected 9000 Bengali datasets from diverse sources, including social media chats, websites, and tech blogs shown inn Figure **??**. The raw data primarily originated from a range of sources such as Facebook group post comments, YouTube comments, Facebook captions, and comments from various blogs and articles. The corresponding Banglish content was written by individuals of different backgrounds to ensure accuracy and capture diverse writing styles for thesis report purposes.

For instance, a single Bengali sentence might be expressed by different users in multiple ways in Banglish. Shown in the table 4.1

kemon achen	kaman acen	keman achen	kmon achen	kamon asen

Table 4.1: Various forms



Dataset_01_NLP(Thesis on Bengali Transliteration)

File Edit View Insert Format Data Tools Extensions Help

C6

	A	B
1	**English transliterated text (Bangla phonetic)**	**Corresponding Bangla Script**
2	"Bismillahir Rahmanir Rahim" Assalamu alaikum.	"বিসমিল্লাহির রাহমানির রাহিম" আসসালামু আলাইকুম.
3	besh upovog korchi	বেশ উপভোগ করছি
4	jar sathe kono karon chhara	যার সাথে কোন কারন ছাড়া
5	ja ache kopale!	যা আছে কপালে !
6	golu ebar joss hoise onek	গুলো এবার জোস হইছে অনেক
7	ghontar por ghonta kotha bola jay	ঘন্টার পর ঘন্টা কথা বলা যায়
8	te asian music hour ekkebare jomiye diche.	তে এসিয়ান মিউজিক আওয়ার এক্কেবারে জমিয়ে দিছে।
9	biday Facebook! valo thako sobai	বিদায় ফেসবুক! ভালো থাকো সবাই
10	184 target nite jai Pakistan 182te all out!	১৮৪ টার্গেট নিতে যাই পাকিস্তান ১৮২ তে অলআউট !!
11	moner vlobasha dio.....	মনের ভালোবাসা দিও.....
12	shesh 17 rune 6wicket!	শেষ ১৭ রানে ৬ উইকেট !
13	shuvo jonmodin ma !!	শুভ জন্মদিন মা !!
14	ami ei page er notun admin	আমি এই পেজ এর নতুন এডমিন
15	valobasha name'e page golo dekhte mone chay na	ভালোবাসা নাম'এ পেজ গুলো দেখতে মনে চায় না .....
16	happy" name'er keu shei womaner kache niye jan,	হ্যাপি" নামের কাউকে সেই উইমেনের কাছে নিয়ে যান,
17	keu kere nite chaile take mrittuboron korbo dan	কেউ কেড়ে নিতে চাইলে টাকে মৃত্যু করবো দান!
18	1 rune haira gelo!	১ রানে হাইরা গেলো!
19	taholei to sha bolbe happy!	তাহলেই তো সে বলবে, হ্যাপি !
20	Jhinuk amar jan,	ঝিনুক আমার জান,
21	age raat hoile ami	আগে রাত হইলে আমি
22	Ar ekhon T L khali	আর এখন টি এল খালি

Figure 4.1: Dataset collection

4.2 Data Cleaning

Eliminate Noise: Remove extraneous characters, symbols, and emojis that might not be helpful for the translation process.

Quality Control: Manually review a portion of the data to find and eliminate any sentences that have poor translations or grammatical problems.

4.3 Tokenization and Subword Segmentation

Tokenizer Selection : Choose a suitable tokenizer that supports the Bengali script. Consider using SentencePiece or Byte-Pair Encoding (BPE) for subword segmentation.

Tokenize: Tokenize both Banglish and Bangla sentences into words or subword units. Ensure that you handle the special tokens, such as <start> and <end>.

4.4 Data Splitting

Shuffling: Shuffle the dataset to ensure randomness during training.

Splitting: Divide the dataset into training, validation, and test sets. A common split is 60% for training, 20% for validation, and 20% for testing.

4.5 Data Augmentation

Paraphrasing: Introduce sentence variations by paraphrasing sentences while retaining the original meaning.

Synonym Substitution: Replace words with synonyms to enhance diversity in your training data.

Handling Imbalanced Data Ensure a balanced representation of various domains and language variations in your training data to prevent biases.

4.6 Data Normalization

Number and Date Normalization: Normalize numbers, dates, and other specific formats to a consistent representation.

Case Normalization: Decide whether to keep the original casing or convert all text to lowercase.

4.7 Bilingual Data Alignment

Verify that each Banglish sentence corresponds accurately to its Bangla translation to maintain alignment in your dataset. And Format your dataset into pairs of input Banglish sentences and target Bangla sentences, ready for training.

Chapter 5

Proposed Approach

5.1 The Methodology

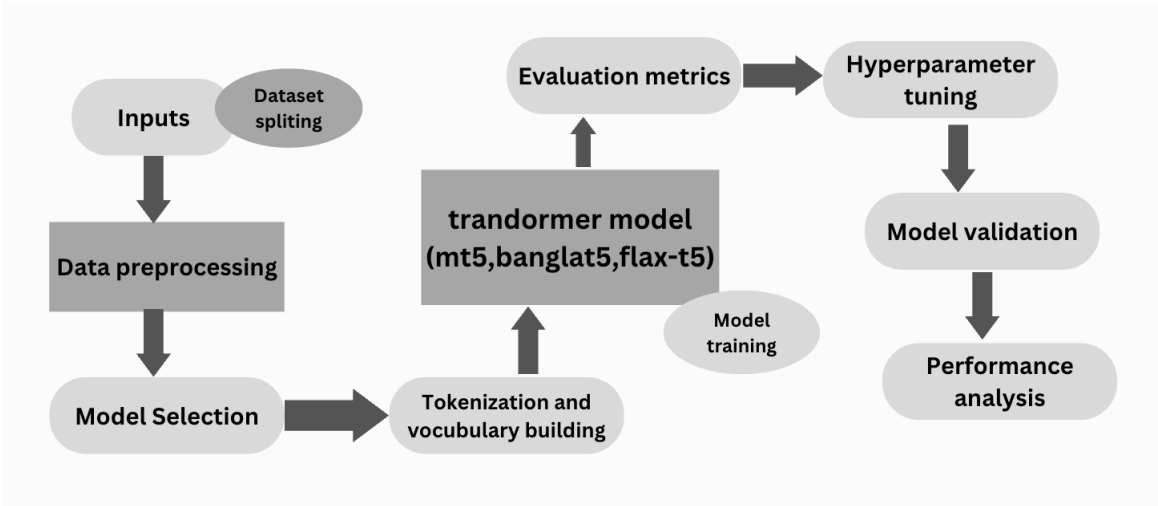


Figure 5.1: The proposed approach

Data Collection, Preprocessing and Splitting

- Collect a sizable parallel corpus containing Banglish sentences paired with their corresponding Bangla translations.
- Preprocess the collected data by removing noise, special characters, and unnecessary formatting. Tokenize the text into subword units using a tokenizer like SentencePiece.

- Divide the preprocessed dataset into training, validation, and test sets using a common split ratio, such as 70% for training, 15% for validation, and 15% for testing.

Model Selection:

- Choose the MT5 (Multilingual Translation with 5 Encoders) transformer model, known for its performance in multilingual translation tasks, to build the translation model. We chose also flax-t5 and bangla t5 model and later compared them.

Tokenization and Vocabulary Building

- Apply tokenization using SentencePiece to convert sentences into subword units, creating a vocabulary specific to Banglish and Bangla languages.

Model Training

- Initialize the model with pre-trained weights and fine-tune it using the Banglish to Bangla parallel corpus.
- Implement teacher forcing during training, feeding the model with the correct target sequence during each step.

Evaluation Metrics

- Assess translation quality using standard metrics such as BLEU, METEOR, and ROUGE.
- Monitor the model's performance on both validation and test sets to avoid overfitting.

These will be discussed in detail in the 6 chapter.

Hyperparameter Tuning

- Experiment with hyperparameters like learning rate, batch size, and dropout rate to optimize the model's translation performance.

Model Validation

- Regularly validate the model's translations on the validation set to ensure effective training progress.

Inference and Translation

- Develop an inference pipeline that takes a Banglish sentence as input.
- Tokenize the input sentence, convert it to subword tokens, and pass it through the MT5 model for translation.
- Convert the model's output subword tokens back to a coherent Bangla sentence using detokenization.

Performance Analysis

- Evaluate the model's translation quality using the chosen evaluation metrics on the test set.
- Conduct a comparative analysis against other machine translation models, if available, to showcase the effectiveness of the proposed approach.

Qualitative Analysis

- Conduct a manual review of translated sentences to identify any linguistic, cultural, or semantic issues introduced during translation.

Results and Discussion

- Present the quantitative results, showcasing the model's performance in terms of translation metrics.
- Discuss qualitative findings, highlighting strengths, limitations, and potential areas of improvement.
- Reflect on the significance of the research and its implications for future work.

5.2 Proposed Model

In this thesis, we choose the google/mt5-base (Multilingual Translation with 5 Encoders) transformer model, known for its performance in multilingual translation tasks, to build the translation model and also we propose models named fax-community/bengali-t5 base, csebuetnlp/banglat5, designed to address the challenges of Bangla language processing across various modalities.

5.2.1 flax-T5 Model

The flax-T5 model is a variant of the T5 (Text-To-Text Transfer Transformer) model developed by Google. It is implemented using the Flax framework, which is built on top of JAX (a library for high-performance numerical computing). The Flax T5 model follows the architecture of the original T5 model, consisting of encoder-decoder transformer layers with shared parameters. It is trained on large-scale text data using a text-to-text approach, where input-output pairs are represented as text strings. The model is capable of performing various natural language processing tasks such as translation, summarization, and question answering, achieving state-of-the-art performance on benchmark datasets.

5.2.2 mt5 Model

Developed by Google, MT5 is a state-of-the-art multilingual translation model trained on a massive dataset covering multiple languages.

- It employs a transformer-based architecture and leverages pre-training objectives such as translation, language modeling, and document classification.
- MT5 demonstrates high performance across various language pairs and tasks, making it a versatile choice for multilingual applications
- mT5 is the multilingual variant of the T5 model pretrained on 101 languages. It has a similar transformer architecture with 2 encoder and decoder layers each, model dimensions being 1024 and 12 attention heads resulting in approximately 770 million parameters. [12]

5.2.3 mt5 Architecture

- Implement the MT5 model architecture, comprising five encoder layers, each specializing in handling different languages and scripts.

Encoder-Decoder Architecture

- MT5 utilizes a transformer-based encoder-decoder architecture, similar to other translation models.

- The encoder processes the source language input, generating contextualized representations.
- The decoder generates the target language output based on the encoder's representations and previously generated tokens.

Multilingual Training

- MT5 is trained on multilingual data, enabling it to translate between multiple language pairs.
- It leverages shared representations across languages to improve translation quality and efficiency.

Fine-tuning for Specific Languages

- After pre-training on multilingual data, MT5 can be fine-tuned on specific language pairs for further optimization.
- Fine-tuning adapts the model's parameters to the characteristics of individual languages, improving translation quality.

Fine-tuning is a process of adjusting the parameters of a pre-trained model to improve its performance on a specific task. In the case of mT5, the pre-trained model is trained on a massive dataset of text and code. This dataset includes books, articles, code, and other forms of text. The model is trained using a technique called masked language modeling, where the model is trained to predict missing words in a text.
Here are some of the steps involved in fine-tuning mT5:

1. Choose a pre-trained mT5 model.
2. Prepare a dataset of text pairs for the task you want to fine-tune for.
3. Choose a fine-tuning hyperparameters.
4. Fine-tune the model.
5. Evaluate the performance of the fine-tuned model.

Cross-lingual Transfer Learning

- MT5 benefits from cross-lingual transfer learning, where knowledge learned from one language can be transferred to improve performance on other languages.
- This allows the model to generalize better across diverse language pairs and tasks.

5.2.4 banglaT5 model

- BanglaT5 is an adaptation of the T5 (Text-to-Text Transfer Transformer) model specifically tailored for the Bengali language created by buetcsenlp.
- It inherits the architecture and pre-training objectives from T5 but fine-tuned on a large corpus of Bangla text.
- BanglaT5 achieves competitive performance in tasks such as text generation, summarization, and question answering, showcasing its effectiveness in Bengali language processing.

5.2.5 banglat5 model Architecture

Adaptation of T5 for Bengali

- BanglaT5 is an adaptation of the T5 (Text-to-Text Transfer Transformer) architecture specifically tailored for the Bengali language.
- It inherits the fundamental architecture and principles of T5 while being customized for Bengali text processing.

Encoder-Decoder Transformer

- BanglaT5 employs a transformer-based encoder-decoder architecture.
- The encoder processes input Bengali text, while the decoder generates output text based on the encoder's representations and previously generated tokens.

Text-to-Text Framework

- Similar to T5, BanglaT5 follows the text-to-text paradigm, where all tasks are formulated as text-to-text transformations.

- This unified framework enables BanglaT5 to handle various natural language processing tasks in Bengali.

Shared Vocabulary and Pre-training

- BanglaT5 uses a shared vocabulary tailored for Bengali text.
- During pre-training, the model learns general-purpose language representations by training on large-scale Bengali text corpora using objectives like denoising autoencoding and masked language modeling.

5.3 Evaluation Metrics

5.3.1 BLEU Score

BLEU (Bilingual Evaluation Understudy) is a metric used to evaluate the quality of machine-translated text by comparing it to one or more reference translations. It measures the similarity between the machine-generated translation and the reference translations based on the overlap of n-grams (contiguous sequences of n words) between them. BLEU score is calculated as a geometric mean of the modified precision scores for n-grams of varying lengths (usually up to 4-grams). A higher BLEU score indicates a better alignment between the machine translation and the reference translations, with scores closer to 1 representing higher quality translations.

5.3.2 Word Error Rate

Word Error Rate (WER) is a metric used to evaluate the performance of automatic speech recognition (ASR) systems or machine translation systems. It measures the rate of errors in the output compared to the reference or ground truth text. WER calculates the number of insertions, deletions, and substitutions needed to transform the output text into the reference text, and then divides this total number of errors by the total number of words in the reference text. Lower WER values indicate higher accuracy, with a WER of 0 indicating a perfect match between the output and reference texts.

The WER formula can be expressed as:

WER = (S + D + I) / N

Where:

S = Number of substitutions

D = Number of deletions

I = Number of insertions

N = Total number of words in the reference text

Validation loss

Validation loss refers to the error or discrepancy between the predicted outputs of a machine learning model and the actual ground truth values on a validation dataset. It is a measure of how well the model is performing during training on unseen data. A lower validation loss indicates better performance and suggests that the model is effectively learning the underlying patterns in the data without overfitting.

Chapter 6

Result and Discussion

6.1 Performance Evaluation Metrics

The following three figures reflects the evaluation metrics in the training session for three models (mt5, flax-t5 and banglat5 model).

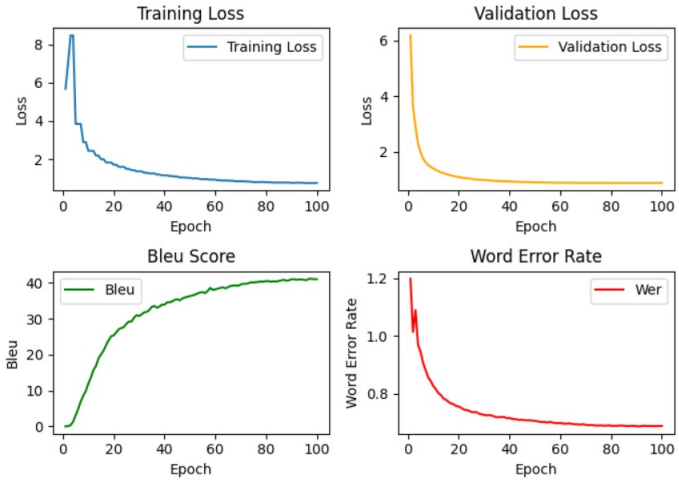


Figure 6.1: Graphical representation of evaluation metrics for mt5

In the provided graph, it's observed that from epoch 1 to 100, the BLEU score shows a consistent increase, indicating improved translation quality over successive training epochs. Simultaneously, both validation loss and training loss exhibit a consistent decrease over the same range of epochs, suggesting that the model's performance improves as it iterates through the training data. Additionally, the word error rate (WER) decreases steadily, indicating enhanced accuracy in the generated

translations. These trends collectively suggest that the model becomes increasingly proficient in its translation tasks as training progresses from epoch 1 to 100.

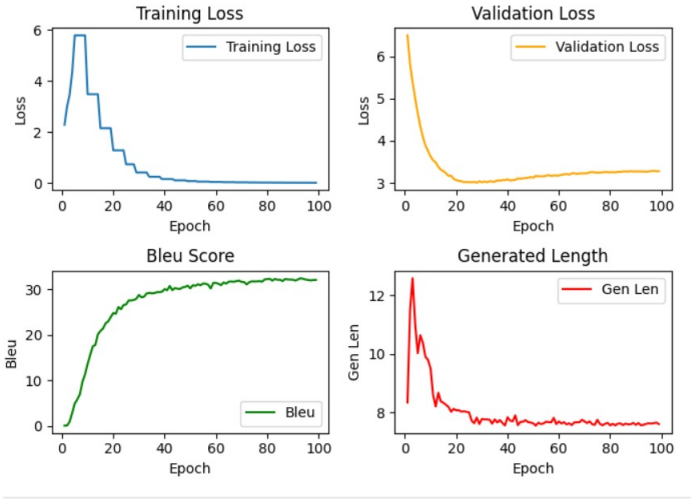


Figure 6.2: Graphical representation of evaluation metrics for flax-t5

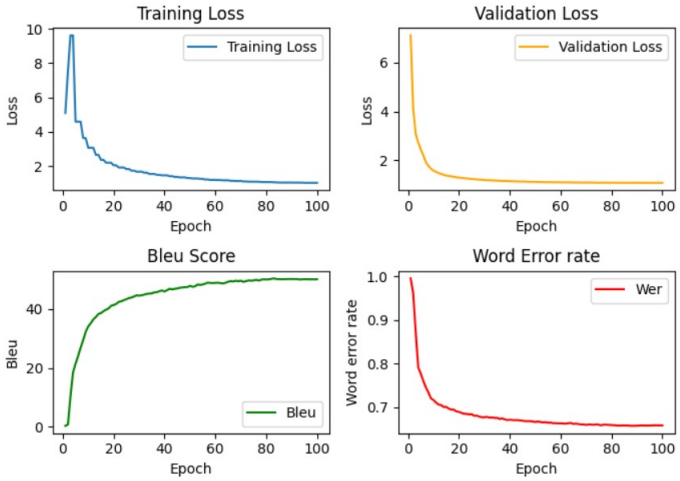


Figure 6.3: Graphical representation of evaluation metrics for banglaT5

6.2 Result Analysis

Model	Dataset	Bleu_{Score}	WER
mt5	Testing(20%)	41.01290261	0.68723
flax-t5	Testing(20%)	32.105500	1.3409
BanglaT5	Testing(20%)	46.0088	0.658049

Table 6.1: Results Analysis

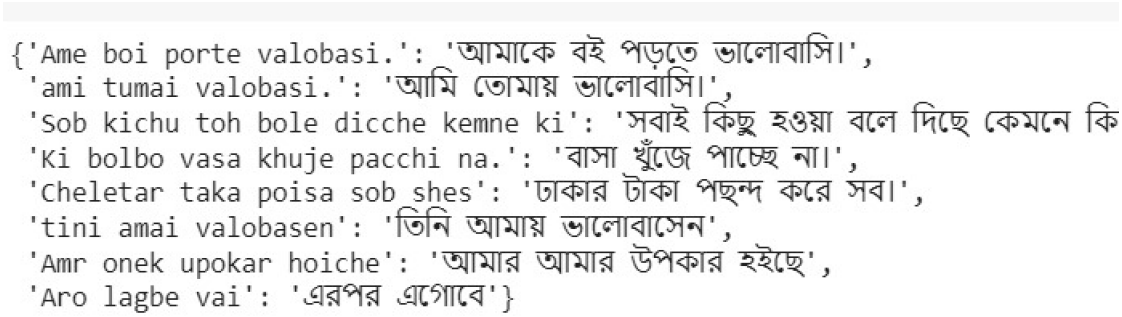


Figure 6.4: Output using mt5

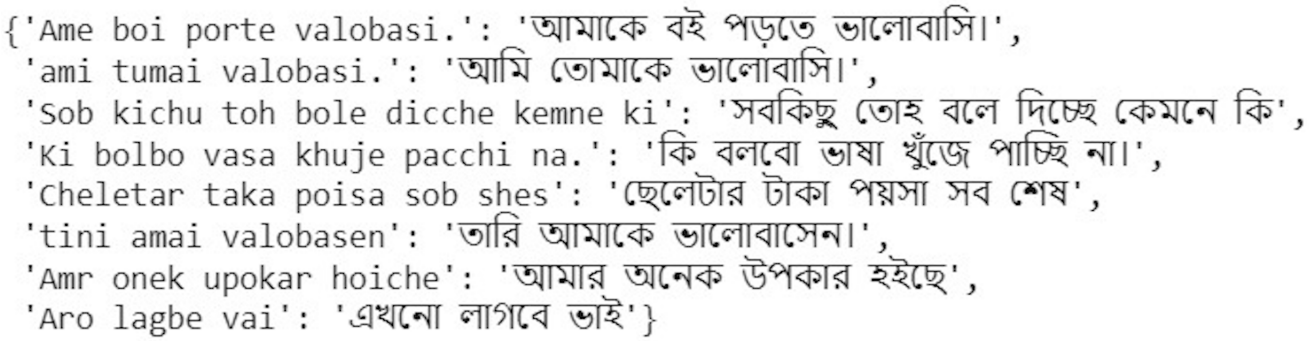


Figure 6.5: Output using bangla t5

In our thesis, we used more than 9000 data for each model and analyze the results obtained from training the three models (MT5, Flax T5, and Bangla T5) using the BLEU scores as follows:

6.2.1 Performance Comparison

- Begin by summarizing the BLEU scores obtained for each model. In our case, MT5 achieved a BLEU score of 41.01, Flax T5 obtained 32.11, and Bangla T5 achieved 46..01.
- Clearly state which model performed the best in terms of BLEU score.

6.2.2 Interpretation of Results

- Discuss the significance of the BLEU scores. For instance, a higher BLEU score indicates better performance in terms of translation quality.
- Analyze why certain models performed better or worse compared to others. Factors such as model architecture, training data size, hyperparameters, and training duration can influence model performance.

6.2.3 Model Strengths and Weaknesses

Highlight the strengths and weaknesses of each model based on the obtained BLEU scores. For example:

- bangla T5, with the highest BLEU score, might excel in multilingual translation tasks due to its architecture and pre-training on diverse bangla language data.
- Flax T5, with a lower BLEU score, might have limitations due to its architecture or training process.
- mT5, despite performing well, may have certain areas where it could be further optimized for better performance.

6.3 Discussion

The widespread use of Banglish (English phonetic) in social chats, comments, blogs, and other forms of social communication highlights its prevalence in digital discourse. However, its inconsistent style can be irritating for some individuals in certain situations. To address this challenge, previous efforts have been made to convert Banglish into correct Bangla form, albeit

with some limitations. In our study, we collected a dataset comprising 9,000 instances of Banglish text and trained it using state-of-the-art models such as MT5, Flax T5, and Bangla T5. While our results show promising improvements, particularly with the bangla-T5 model, it is evident that our dataset size remains a limiting factor in achieving optimal translation quality. Thus, there is a clear need to expand the dataset to enhance translation accuracy and performance further. Our findings underscore the importance of continued research and development in this area to refine translation systems and address the challenges posed by Banglish communication. By expanding the dataset and leveraging advanced machine learning techniques, we can strive to create more effective solutions for translating Banglish to Bangla, thereby improving communication experiences for diverse users in digital environments.

Chapter 7

Conclusion

In our thesis, we explored the task of Banglish to Bangla machine transliteration using advanced machine learning models, including MT5, BanglaT5, and FlaxT5. Through the collection and training on a dataset comprising 9000 instances of Banglish text, we aimed to address the challenge of accurately transliterating Romanized Bengali text into its native Bengali script. Our experiments with the MT5, BanglaT5, and FlaxT5 models demonstrated promising results, showcasing their effectiveness in capturing the nuances of Banglish and producing accurate transliterations into Bengali. The models exhibited improvements in transliteration quality as evidenced by the evaluation metrics such as BLEU score, validation loss, training loss, and word error rate (WER). Specifically, we observed consistent increases in BLEU scores, indicating enhanced translation quality, while simultaneously witnessing reductions in validation loss, training loss, and WER, indicative of improved model performance and accuracy. These findings underscore the potential of leveraging state-of-the-art machine learning models for Banglish to Bangla transliteration tasks. In conclusion, our thesis contributes to the advancement of machine transliteration systems for Bengali language processing, providing valuable insights and methodologies for future research and development in this domain. The success of our experiments underscores the importance of utilizing advanced machine learning techniques and large-scale datasets for achieving accurate and reliable Banglish to Bangla transliteration. As technology keeps changing, how we use languages will too. Future work could look at using our system in different apps to make it easy for people to use many languages.

References

[1] J. Kaur, "Machine transliteration system in indian perspectives," *International Journal of Science, Engineering and Technology Research (IJSETR)*, vol. 2, no. 5, 2013.

[2] N. T. Le and F. Sadat, "Low-resource machine transliteration using recurrent neural networks of asian languages," in *Proceedings of the Seventh Named Entities Workshop*, 2018, pp. 95–100.

[3] N. Paul, I. Faruki, M. I. Pranto, M. T. Rouf Shawon, and N. C. Mandal, "Bengali-english neural machine translation using deep learning techniques," in *2023 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2023, pp. 1–6.

[4] Z. Islam, J. Tiedemann, and A. Eisele, "English to bangla phrase-based machine translation," in *Proceedings of the 14th Annual conference of the European Association for Machine Translation*, 2010.

[5] M. Khairullah, "A novel steganography method using transliteration of bengali text," *Journal of King Saud University-Computer and Information Sciences*, vol. 31, no. 3, pp. 348–366, 2019.

[6] N. UzZaman, "Phonetic encoding for bangla and its application to spelling checker, transliteration, cross language information retrieval and name searching," Ph.D. dissertation, BRAC University, 2005.

[7] N. AbdulJaleel and L. Larkey, "English to arabic transliteration for information retrieval: A statistical approach," *Center for Intelligent Information Retrieval Computer Science, University of Massachusetts*, 2003.

[8] G. Wei, "Phoneme-based statistical transliteration of foreign names for oov problem," *Master's Thesis, The Chinese University of Hong Kong*, 2004.

[9] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech communication*, vol. 50, no. 5, pp. 434–451, 2008.

[10] M. M. Hossain, M. F. Labib, A. S. Rifat, A. K. Das, and M. Mukta, "Auto-correction of english to bengali transliteration system using levenshtein distance," in *2019 7th International Conference on Smart Computing & Communications (ICSCC)*. IEEE, 2019, pp. 1–5.

[11] G. S. Shibli, M. T. R. Shawon, A. H. Nibir, M. Z. Miandad, and N. C. Mandal, "Automatic back transliteration of romanized bengali (banglish) to bengali," *Iran Journal of Computer Science*, vol. 6, no. 1, pp. 69–80, 2023.

[12] V. Agarwal, P. Rao, and D. B. Jayagopi, "Hinglish to english machine translation using multilingual transformers," in *Proceedings of the Student Research Workshop Associated with RANLP 2021*, 2021, pp. 16–21.

[13] K. Knight and J. Graehl, "Machine transliteration," *arXiv preprint cmp-lg/9704003*, 1997.

[14] N. AbdulJaleel and L. S. Larkey, "Statistical transliteration for english-arabic cross language information retrieval," in *Proceedings of the twelfth international conference on Information and knowledge management*, 2003, pp. 139–146.

[15] A. Finch and E. Sumita, "Transliteration using a phrase-based statistical machine translation system to re-score the output of a joint multigram model," in *Proceedings of the 2010 Named Entities Workshop*, 2010, pp. 48–52.

[16] K. Deep and D. V. Goyal, "Hybrid approach for punjabi to english transliteration system," *International Journal of Computer Applications*, vol. 28, no. 1, pp. 0975–8887, 2011.

[17] A. Das, A. Ekbal, T. Mondal, and S. Bandyopadhyay, "English to hindi machine transliteration system at news 2009," in *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration (NEWS 2009)*, 2009, pp. 80–83.

[18] S. Mohanlal, B. Sharada, A. Fatihi, L. Gusain, K. Karunakaran, and J. M. Bayer, "A proposal for standardization of english to bangla transliteration and bangla editor joy mustafi, mca and bb chaudhuri, ph. d." *Language in India*, vol. 8, no. 5, 2008.

[19] M. Rosca and T. Breuel, "Sequence-to-sequence neural network models for transliteration," *arXiv preprint arXiv:1610.09565*, 2016.

[20] S. Kundu, S. Paul, and S. Pal, "A deep learning based approach to transliteration," in *Proceedings of the seventh named entities workshop*, 2018, pp. 79–83.

[21] T. Deselaers, S. Hasan, O. Bender, and H. Ney, "A deep learning approach to machine transliteration," in *Proceedings of the Fourth Workshop on Statistical Machine Translation*, 2009, pp. 233–241.

[22] M. M. Mahsuli and R. Safabakhsh, "English to persian transliteration using attention-based approach in deep learning," in *2017 Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 2017, pp. 174–178.

[23] A. Al Taawab, L. Tasnia, M. Dhar, and M. H. K. Mehedi, "Transliterated bengali comment classification from social media," in *2022 IEEE 10th Region 10 Humanitarian Technology Conference (R10-HTC)*. IEEE, 2022, pp. 365–371.