Introducing A Bangla Sentence–Gloss Pair Dataset for Bangla Sign Language Translation and Research

by

Nafis Ashraf Roudra
21301410
Neelavro Saha
21301181
Rafi Shahriyar
21301198
Saadman Sakib
21101091

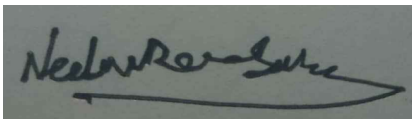
Department of Computer Science and Engineering
School of Data and Sciences
Brac University
June 2025

Declaration

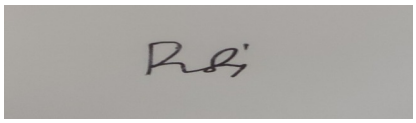
It is hereby declared that

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

Students' Signatures, Names, and IDs:

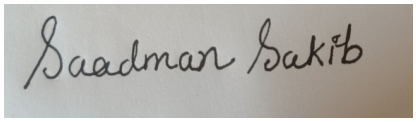


Neelavro Saha
21301181



Rafi Shahriyar
21301198

Nafis Ashraf Roudra
21301410



Saadman Sakib
21101091

Approval

The thesis titled "Introducing A Bangla Sentence–Gloss Pair Dataset for Bangla Sign Language Translation and Research" submitted by:

1. Nafis Ashraf Roudra (21301410)
2. Neelavro Saha (21301181)
3. Rafi Shahriyar (21301198)
4. Saadman Sakib (21101091)

Examining Committee:

Supervisor:
(Member)

Annajiat Alim Rasel
Senior Lecturer
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Md Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson
Associate Professor
Department of Computer Science and Engineering
BRAC University

Abstract

Bangla Sign Language translation and recognition has been an evolving research topic throughout the years. However, existing research on this field is limited to word and alphabet level detection. For a more continuous sentence level detection of spoken Bangla sentences and their corresponding signed gestures, structured and comprehensive annotations are essential. Therefore, in this paper we introduce a dataset that consists of Bangla sentences and their matching gloss sequence pairs. Gloss sequences are made up of individual glosses which are Bangla sign supported words and serve as an intermediate representation for a continuous sign. Our dataset consists of 1000 high quality Bangla sentences that are manually annotated into a gloss sequence by a professional signer. With no available open-source datasets on Bangla sentence and gloss pairs, we additionally augment our dataset with 3000 synthetic samples. For the augmentation process, we introduce a RAG-based pipeline which incorporates rule-based linguistic strategies and prompt engineering techniques that we have adopted by critically analyzing our human annotated sentence-gloss pairs and by working closely with our professional signer. Furthermore, we finetune several transformer-based models such as mBart-50, Google mT5, GPT4.1-nano and perform BLEU score based evaluations to determine which model performs the best in the task of Sentence-to-gloss translation. Finally, based on these evaluation metrics we also compare how our dataset performs against the Phoenix-2014T dataset.

Keywords: Natural Language Processing; Transformers; Machine Translation; Bangla Sign Language; BdSL; Gloss Translation; Data Augmentation; RAG; Low-resource Languages;

Table of Contents

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$AddSub$ Addition and Subtraction Dataset

$AQUA-RAT$ Algebra Question Answering with Rationales

ASL American Sign Language

$Auto-CoT$ Automatic Chain-of-Thought

$BART$ Bidirectional and Auto-Regressive Transformers

$BERT$ Bidirectional Encoder Representations from Transformers

$BLEU$ Bilingual Evaluation Understudy (text generation metric

CMR Conditional MoE Routing

$CoNLL03$ Conference on Natural Language Learning 2003

CoT Chain-of-Thought

$CSQA$ Commonsense Question Answering

DDP Distributed Data Parallel

DPR Dense Passage Retriever

EOM Expert Output Masking

$FEVER$ Fact Extraction and VERification

FFN Feed-Forward Network

$fp16$ 16-bit Floating Point Precision

$fp32$ 32-bit Floating Point Precision

GCN Graph Convolutional Network

$GPT-3$ Generative Pre-trained Transformer 3

$GSM8K$ Grade School Math 8K

$LAMBADA$ LAnguage Modeling Broadened to Account for Discourse Aspects

$Manual - CoT$ Manually Constructed Chain-of-Thought

$mBART$ Multilingual Bidirectional and Auto-Regressive Transformer

$MetaST$ Meta Self-Training

MoE Mixture of Experts

$MSMARCO$ Microsoft MAchine Reading COmprehension

$MultiArith$ Multiple Arithmetic Problems Dataset

$NLLB$ No Language Left Behind multilingual translation model

NMT Neural Machine Translation

PLM Pretrained Language Model

$PromDA$ Prompt-based Data Augmentation

QRA Question-Rationale-Answer

RAG Retrieval-Augmented Generation

$ROUGE - L$ Recall-Oriented Understudy for Gisting Evaluation

RT Rotten Tomatoes

$SDANER$ Semantic Data Augmentation for Named Entity Recognition

$SingleEq$ Single Equation Problems Dataset

$SST - 2$ Stanford Sentiment Treebank

$StrategyQA$ Strategy Question Answering

$SVAMP$ Single Variable Arithmetic Math Problems

$T5 - Large$ A large-scale pretrained text-to-text transformer model

$WikiAnn$ Wikipedia-based Annotation for Named Entities

mT5 Multilingual Text-to-Text Transfer Transformer

T5 Text-to-Text Transfer Transformer

Chapter 1

Introduction

Developments in Natural Language Processing has made multiple spoken languages more accessible through machine translation. Following the introduction of Transformer based architectures, models can be trained to exhibit enhanced multilingual understanding, allowing information to be distributed despite linguistic barriers [4] [10]. However, even in today's world, there still remains a huge accessibility challenge for the deaf community - particularly when it comes to reading and understanding text or spoken language. Deaf people prefer sign language as a primary communication method. However, this mode of communication is not incorporated in much of the content of the modern day, which are mostly voice or text-based. Research shows that when people read, they associate words with their corresponding pronunciations in order to comprehend the text [8]. Deaf people, unable to hear, lack this phonological skill as they have no idea what the words sound like. As such, reading text is challenging for the deaf population. To tackle these issues, ongoing research include creating pipelines of different deep-learning and transformer based models to convert spoken text to its corresponding sign language translation in an animated form [22] [28]. These pipelines require extensive datasets containing spoken sentences, their corresponding videos, and gloss sequences that serve as a type of structured annotations that represent sign language in textual form. While text-to-gloss translation has been explored in various languages globally, little work has been done in the context of Bangla. In Bangladesh, there are approximately 13 million people with hearing difficulties, 3 million of whom are deaf [1]. Many of these people use Bangla sign language (BdSL) as the primary method of communication. Therefore, introducing a dataset of Bangla sentence and gloss sequence pairs could serve as a foundational step for facilitating research on comprehensive Bangla sign language translation systems and thus ensuring more consistent accessibility for the deaf population.

1.1 Problem Statement

Sign language is a primary method of communication for many individuals in the deaf population. However, most information is not available in this mode of communication, rather in the form of voice or text. Deaf people have difficulty comprehending written text due to their inability to make phonological connections with pronunciations of the words [8]. This warrants the need for a system that converts text to structured sign language representations. There have been several works in the field of text to sign translation, aiming to improve information acces-

sibility of deaf people [28] [13] [17]. However, most of these works do not address the Bangla sign language community - they are based on high resource languages like English, German or Chinese. In the field of Bangla language and BDSL, most existing research focuses on sign language to text translation - particularly on the alphabet-level ("অ", "আ", "ই") [19] [24] and word-level ("আমি", "আমরা", "তুমি") conversion [26] [23]. While these works help non-sign language users to understand sign language, it does not work the other way around i.e. it does not aid deaf people in accessing non-sign language information such as voice or text. As such, this is only one side of the coin and it does not fully address the accessibility problem faced by deaf individuals. To tackle this problem comprehensively, a robust system for translating Bangla text to BDSL gloss sequences is required, but research in this area is very limited. Existing works [14] [18] are limited to the translation of a few words or numbers - such as, "১", "২", "৩", "আমি", "তুমি" etc - into basic sign representations. However, these works do not address translation of continuous Bangla text, such as - "আমি ভাত খাই" , which leaves much room for improvement. Moreover, another key problem in translating from Bangla text to BdSL is that they have different grammatical structures [2]. "আমার পাঁচটি পেন আছে" translates to "আমার পেন পাচ" in BdSL gloss representation. This means that we cannot directly convert Bangla text into sign language without an intermediary an intermediary step that addresses this translation. There exists no comprehensive dataset for translation of Bangla text to BDSL gloss sequences. Consequently, research has been limited to a few words and phrases. Therefore, creating a high-quality annotated dataset of Bangla text-to-gloss pairs would address this fundamental gap and enable the development of more sophisticated sign language translation systems.

1.2 Aims and Objectives

Through the introduction of our expert annotated high-quality Bangla Sentence-Gloss pair dataset, this work aims to lay down the foundation for further research into Continuous Bangla Sign Language Recognition and translation, a field that remains vastly unexplored due to a severe lack of publicly available resources. Gloss sequences act as a bridge that connects natural language to sign language, so they serve a crucial role in Sign Language NLP tasks. Moreover, these gloss sequences can also be used as an intermediary step towards 3d animation of sign language sentences. These animations can then be integrated into real-world applications, such as education tools or virtual interpreters, significantly benefiting deaf individuals by enhancing their interaction in both academic and social environments.

Chapter 2

Literature Review

2.1 Bangladeshi Sign Language (BDSL)

In this paper, Rubaiyeat et al. [26] introduced BdSL60, an extensive and one of the very few word-level Bdsl dataset consisting of 60 annotated Bangla sign words and 9307 video trials from 18 signers. Various preprocessing methods were applied to the dataset. A PROLONGED version of the dataset was created by duplicating frames to maintain uniformity for machine learning models. A FLIPPED version was also created by transforming all left-handed sign instances to right-handed. A Relative Quantization (RQ) dataset was also created by quantizing the landmark points to reduce depth variations. Overall, the dataset variations boil down to: i) PROLONGED and FLIPPED, ii) PROLONGED and NON-FLIPPED, iii) NON-PROLONGED and FLIPPED, iii) NON-PROLONGED and NON-FLIPPED and finally v) RQ. Mediapipe Hollistic was used to extract pose keypoints and traditional SVM, SVM-DTW, and Attention-based Bi-LSTM were used to benchmark the dataset. Classical SVM provided a testing accuracy of about 67.5%, while SVM-DTW scored around 65.8%. On the other hand, attention-based biLSTM had a testing accuracy of up to 75.1%.

Sams et al. [23] introduced SignBD-Word a word-level dataset in video format for Bangla sign language. The dataset contains 6,000 video clips of 200 distinct Bangla sign language words with each sign recorded from both full-body and upper-body perspectives. Additionally, annotations and glosses are provided for each video. Collected with the help of 16 different signers Sams et al. [23] provides a dataset that serves as a baseline for future research in BdSL recognition and pose synthesis. According to this paper the authors used OpenPose to draw out the 2D key-point features from the body present in the frame. The authors then employ VGG-19, a deep convolution neural network architecture, to understand the relationships between different body parts and also identify specific body parts such as hands, elbow and shoulder. The output from the VGG-19 model consists of confidence maps and Part Affinity Fields (PAFs). The confidence maps represent the likelihood of correctly detecting specific body parts, while the PAFs capture the spatial relationships between them.These outputs are then processed using a bipartite matching algorithm. This algorithm effectively assembles the detected body parts into a coherent skeleton-like posture. To standardize the dimensions of all sign language video clips, individual videos were sampled at 30 frames and at consistent intervals. Individual frames were then re-scaled to 224×224 pixels using bi-linear

interpolation. Afterwards they employ CNN-LSTM, CNN-Attention-LSTM, I3D and SlowFast on the skeleton body pose and RGB video data for the sign language classification task. Results indicate that CNN-LSTM and CNN-Attention-LSTM models do not achieve satisfactory performance when it comes to detecting the correct sign. Contrary to, 3D CNN models such as I3D and SlowFast outperform 2D CNN models significantly. I3D on body pose datasets earns a top-1 score of 65.53 and 51.5 for the SignBD-Word-90 and SignBD-Word dataset but if top-5 accuracy is considered it achieves a scores of 89.02 and 79.5 for the corresponding datasets. Similarly for SlowFast the top-1 accuracy scores for the two datasets are 57.38 and 47.6. The top-5 scores are 85.03 and 75.92. For the RGB videos in SignBD-Word-90 and SignBD-Word datasets SlowFast gets a top-1 score of 66.05 and 57 on the respective datasets comparably top-5 scores are of 88.82 and 84.17 on the datasets. I3D falls short by a small margin in both the top-1 and top-5 accuracy scores. Subsequently the authors explored GAN-based methods for synthesizing sign languages videos from the 2D body-pose skeletons. Three models were used they are as follows, CycleGan, pix2pixHD and SPADE. CycleGAN model produced blurred and noisy results correspondingly SPADE also struggled due to its reliance on semantic label maps. Therefore, pix2pixHD stood out as the best as outputs consisted of clear hand and face details. Realism of the synthesized videos were determined through a Mean Opinion Score (MOS) test. Here 20 university students rated the generated videos. Out of the 3 models pix2pixHD was rated the highest. The primary limitations of this study includes the limited number of signers, the restricted set of sign words, and the need for further fine-tuning of the GAN models for better performance.

2.2 Sign Language Datasets

Li et al. [5] introduced a large dataset for American Sign Language (ASL) recognition for training robust deep learning models. Sign language recognition is a challenging task due to its multimodal nature which involves continuous hand movement, different body orientations, handshapes and facial expressions. As such, a large scale dataset is required to enable deep learning techniques for sign language recognition. This was the motivation behind this paper as it tries to bridge the gap between ASL recognition and current computer vision advancements. The dataset, MS-ASL, has over 25 thousand videos of approximately 1000 ASL signs. Moreover, it has 222 distinct signers, and the videos were shot in an uncontrolled environment, under real-life conditions with variations in lighting and background. These videos were collected from public sources like educational ASL videos from YouTube. Subsequently, the signs were processed to segment the signs into individual samples. OCR was used to extract text from the videos and the subtitles were used for metadata. Then it was reviewed manually to ensure quality, particularly by cropping the videos that are too long. Also, synonyms with ASL vocabulary were handled manually. Afterwards the dataset was divided into train, test and validation sets containing videos of 165, 20 and 37 signers respectively, and state of the art models like I3D, 2D-CNN-LSTM and body key-point recognition were used to benchmark the dataset. Among these models, I3D had the best results as it showed an accuracy of 57.69%, while the other models had much lower accuracy. I3D also had the best results in the top-five accuracy results, as it scored 81.08% in that metric. While this large scale dataset offers significant advancements in sign language recognition, it

has some limitations. There were not sufficient samples for some of the signs which resulted in these classes having lower performance. Additionally, some non-native annotators were involved in the manual annotation process which may introduce some errors. Furthermore, the dataset does not address regional variations in ASL, which may affect the generalizability of the models.

2.3 Linguistic and Gloss sign analysis using Transformer model

Sinha et al. [27] describes a unique approach of converting spoken language text to sign language gloss. They have employed NMT and GCN in their approach. Their method considers embeddings, i.e., vector representations based on the words in the given text, as well as incorporating additional interpretation retrieved from the arrangement of the input. The ASL allows the speaker to focus more on a particular expression while communicating, altering the overall sequence of the whole message being delivered without causing much distortion. This pliability contributes to the feasibility of deducing a fixed rule-based method for translation. However, the current NMT methods struggle to retrieve important messages conveyed during communication as they miss intricate linguistic details. The model described in this paper utilizes Word Piece Tokenization, fragmenting the text and converting the consequent words into tokens. Each token in the sequence is then converted to a vector representation called Token Embedding. There is another component useful for locating the positions of the token in the series. All these three embeddings are added in order to retrieve the total input through the expression(2.1):

$$h_i \in \mathbb{R}^{(n+2)\times h} \tag{2.1}$$

where n is the total number of tokens and h is the latent size of the models.
The GCN involves representing the entire text through a Syntactic Dependency Graph where the words are represented via the vertices of the graph, and the edges resemble the semantic correlation among the words in that text. In this case, a convolution operation is executed on each vertex, which involves retrieving information regarding that central vertex and its surrounding vertices. This enables us to deduce the characteristics of each of the central vertex and the impact of the surrounding vertices in its behavior. The latent states of the GCN operation as output are combined with the embeddings obtained by the encoder of the transformer model and then fed to the decoder. The datasets used in this paper are ASLG-PC12 Corpus and ASL-Bank Dataset (focusing on the banking domain). Fine-tuning was carried out on the ASLG-PC12 Corpus, and then in order to adjust to a particular domain (i.e., banking), ASL-Bank is brought into the picture. For evaluation, they have used ROUGE-L and BLEU scores. BERT has been pre-trained with English language, and hence it is not likely to fit well with gloss arrangements. Therefore, they have accumulated individual gloss words to create full gloss sentences, which are then evaluated using cosine similarity calculation. The ROUGE-L computes the longest common subsequence to estimate the extent to which the predicted gloss preserves the original sequence. The BLEU determines the quality of the translation outputs. The transformer model gave better output for text-to-gloss conversion compared to the traditional GRU approach for both datasets. The main limitation

of this paper is that it is mainly centered upon ASL only, and thus not applicable for other different types of sign languages. The videos that will be generated from the glosses will need careful synthesis; otherwise, inconsistencies may arise while generating continuous ongoing video outputs, and this paper does not shed light on that part.

Another prominent approach implemented by Xuan et al.[29], confronts the issues that arise when NMT is integrated in sign-language to gloss conversion process. The major issue to be addressed is that the available data do not always represent the domain properly and thus causing the model to perform poorly. Another problem is the inconsistency of the synthetic data that was generated by the translation of spoken-language-text to gloss, and used to supplement the training data. Therefore, the paper's work is focused on the betterment of the process of converting sign language glosses to spoken language text - the reversed process of the earlier methods discussed in the literature review. The work in this paper has been fragmented into three different segments i.e., back translation, curriculum learning, and data selection. The back translation is used to translate the intended language back to the initial language, creating a synthetic set of data. Curriculum learning is used to escalate the complexity of the model systematically, enabling it to learn at an efficient rate. Data selection is implemented by selecting superior-quality sign-language data which are befitting to the area of the parallel data. This is done to compensate for the problems due to dataset shift. The conditional probability of creating the intended text line given the equation(2.2):

$$p(y|x) = \prod_{j=1}^{J} P(y_j \mid y_{<j}, x, \theta) \tag{2.2}$$

where θ denotes the parameters of the model, J represents the target sentence length, y_j is the jth target word, and $y_{<j}$ is the prefix of the word before y_j.
The datasets used in this paper are Phoenix-14T, a sign language dataset with two parts for evaluation, and CSL Daily, which is used for comparing and evaluating translation models. The method in this piece of work is heavily dependent on the quality of synthetic data, which causes lower quality data to yield unreliable results. Incongruity in domains might cause problems as well, such as bad performance outcomes when the monolingual data often do not fit well with the parallel data. The model's capability of learning can be affected by the non-uniform representation of data.

Camgoz et al. [3] shed light on sign language to a greater depth, along with its intricate linguistics and sentence structures. The authors sought to cover most of the research gaps that are prevalent in this field of study. Therefore, the aim of this paper is to ensure a significant extent of understanding sign language and its corresponding translation. Several models have been implemented in this paper, and they are all subsets of Neural Network Architecture. The mBART model was used for translational purposes, enabling translational tasks with multilingual data. The CNN-based spatial embedding is used to capture the dimensional characteristics of the sign language videos. The RNN-HMM hybrids were used for token formation and modeling sequences. The encoder-decoder networks based on attention facilitate synchronization between input and output.The mathematical tokenization process

is expressed here in the following expression (2.3):

$$z_{1:N} = \text{Tokenization}(f_{1:T}) \tag{2.3}$$

The dataset used in this experiment was RWTH-PHOENIX-WEATHER 2014, which consists of segments of videos, gloss representations, and the designated translations. The S2G2T model yielded promising results. The mBART model, due to its capability of working with multiple language data, contributes greatly to high accuracy. The main limitation is that there is no previous research cited in this paper. There is no direct correlation between the sign values given as inputs and the text retrieved as output. In addition, the data collection process, as mentioned in this paper, is very tedious and requires extensive preprocessing.

2.4 Prompting and Retrieval-Augmented Techniques in NLP

A quite intriguing approach introduced by Lewis et al. [9] focus on a combined architecture leveraging the robustness of both parametric as well as non-parametric memory to enhance the performance of NLP tasks which are highly knowledge-intensive. The model constitutes a pretrained generator i.e. seq2seq (BART-Large with a parameter count of 400M) along with a neural retriever which is created using two BERT encoders i.e. DPR. BART, in this case acts as a parametric memory storing an overall linguistic idea. DPR, on the other hand, fetches updated information by looking up a big corpus, for example Wikipedia. The core of innovation in this case is the connected full-length fine-tuning of the retriever as well as the generator using a hidden variable technique, where the fetched documents are used as hidden elements i.e. z. Since the fetched document is already hidden, the model instead of choosing a document, calculates the probability associated with the output by adding up all the fetched top-k documents, and the model assigns greater weight to the document with greater relevance. Hence, this method trains the retrieving element to collect documents from the large corpus that aids the generator to generate good-quality output as well as training the generator to make efficient use of the collected documents. Therefore the two components are unified and trained as a single model to ensure significantly better coaction between them. In the following cases the two following models are observed, the first being the RAG-Sequence, where the model chooses a single document among top-k and uses it for the generation of the whole output all throughout the same sequence, presented in equation(2.4).

$$p_{\text{RAG-Sequence}}(y \mid x) \approx \sum_{z \in \text{top-}k} p_\eta(z \mid x) \prod_i p_\theta(y_i \mid x, z, y_{1:i-1}) \tag{2.4}$$

The second one involves dynamically choosing a document based on its reliability among the top-k for one single given token of the text, thus introducing enhanced variation as well as flexibility, presented in the following equation(2.5)

$$p_{\text{RAG-Token}}(y \mid x) \approx \prod_i \sum_{z \in \text{top-}k} p_\eta(z \mid x)\, p_\theta(y_i \mid x, z, y_{1:i-1}) \tag{2.5}$$

RAG gave profound outputs throughout four QA benchmarks that are open-domain which includes Natural Questions, CuratedTrec, TriviaQA, WebQuestions - exceeding the performance of DPR and REALM by 4 to 6 points of EM. RAG-Sequence

did better than BART with +2.6 BLEU and ROUGE-L in MSMARCO. Evaluations by human reviewers show that RAG-Token generates more authentic questions compared to BART in more than 40% of dataset examples from Jeopardy. In the case of FEVER, the findings on RAG nearly makes it the state-of-art model even in the absence of any retrieval in a supervised environment.

Wang et al. [20] introduces the concept of PromDA, a data augmentation method based on prompts intended for low-resource Natural Language Understanding Tasks i.e. classification of sentences and labeling of sequences. The core novelty resides in applying the tuning procedure on soft prompts i.e. special learnable vectors in multiple layers instead of entirely fine-tuning the large Pretrained Language Model. This method enables avoiding the overfitting or memorisation of the small dataset when subjected to a much larger pretrained model like (T5-Large), they can also generate good quality, various synthetic data and also reduce the need for additional unlabeled data from the same domain. The whole pre-trained language model (T5-Large) is frozen, which means the weights do not get changed during training, thus avoiding the fine-tuning of the model itself. In this case, the small learnable vectors i.e. the soft prompts are assigned at all the transformer layers. Prompt parameters are used, updated with the aid of Adafactor optimizer having $1\times10-3$ along with 1000 steps on data selected for few-shot and increased up to 5000 steps for data with greater number of shots. The total number of parameters trained in this technique is around 246K which is significantly lower than fine-tuning the entire model. The method constitutes a dual-view production where there is an input perspective and output perspective. The input perspective constitutes the use of certain selected keywords from the main text to generate a complete sentence based upon those keywords, reflecting the common writing practice based on certain keywords. On the other hand, the output perspective constitutes the use of label types such as categories, and thus generating an accurate sentence falling under that particular label; this is done in order to add variation among the sentences. These sentences from two separate views are then combined to form sentences which are rectified via iterations, along with getting rid of certain samples thus alleviating performance of the model. The PromDA method, in the case of tasks involving Sequence Labeling, performed better above the baselines, in 10 shot settings having a value of +4.8% for CoNLL03 and +7.5% for Wikiann. This method is also better than rule-based techniques like SDANER and MetaST as well as completely fine-tuned PLMs including LAMBADA particularly in the presence of smaller datasets. In case of tasks involving sentence classifications, PromDA shows an enhanced performance for both 10 shot and 50 shot scenarios, increasing the value 66.1% up to 81.4% in SST-2 and then 57.8% up to 73.4% in RT. Therefore, in brief this method yields a profound generalization for large pretrained models over smaller datasets. However one major limitation of this approach is that it is highly reliant on the PLM that is being used as if the PLM has biased attributes the model's performance will be impacted. Also the filtering method of getting rid of lower quality texts could also get rid of samples that could add diversity to the synthetic data.

In recent times, for yielding multiple stage reasoning in Large Language Models, Chain-of-Thought(CoT) prompting has turned out to be one of the most prominent methods. However, the CoT prompting done manually i.e. generating QRA triplets

using first-hand methods is time consuming, constrained within a certain domain and susceptible to unintended biases by humans. In this case, Zhang et al. [21] has introduced Auto-Cot, an approach to automatically create varieties of chains incorporating reasoning with the aid of the LLMs on their own without any external manual stimulation by humans. There are two stages in this approach- first being the question clustering method where the dataset is split to form clusters to create variations. From one cluster of closely-related questions, a single question is selected based on its ability to encompass all the core features of the questions residing in the following cluster. For that particular question, the chain of reasoning is produced with the aid of CoT prompts using Zero-shot i.e. taking one step after another gradually with the help of GPT-3. The resulting illustrations are then incorporated in in-context learning to answer new questions whose answers are not known and to be determined with the aid of few-shots.The parameters are mainly the cluster count, k as well as templates used for prompting which constitutes instructions stepwise generation. The datasets used for arithmetic reasoning are MultiArith, AddSub, AQUA-RAT, SingleEq, SVAMP, GSM8K , the datasets used Commonsense reasoning are CSQA, StrategyQA and for symbolic reasoning they have used Last Letter Concatenation and Coin Flip. Auto-Cot either matched or outperformed all the Manual stimulations for all the chosen standard datasets. In the incorporation of GPT-3, Auto-CoT exceeded the prompting approaches that are randomized, illustrating the former's capability in varying tasks based on reasoning. In the case of GSM8K, Auto-CoT yields an accuracy value of 62.8 % , exceeding Zero-Shot which yielded 31.8% accuracy as well as the Manually stimulated chain which yielded 59.4%. In the case of MultiArith, Auto-CoT yields an accuracy value of 93.2 % almost getting close to the Manual's 96.8%. In the case of AddSub, Auto-CoT again yielded 91.9%, thus getting ahead of both Manual-CoT's 84.6 % as well as Zero-Shot's 65.6%. Hence all these findings indicate the proficiency of this model in terms of diversity in the generated data as well as the reduction of human efforts as well as biases.

Chapter 3

Methodology with workflow

Firstly, we created a foundation dataset of 1000 Bangla sentences manually annotated into BDSL gloss sequences by a professional signer. These sentences were collected from a variety of sources, including Bangla newspaper corpora, bangla literature, BDSL language development book etc. Following the annotation process, we collaborated with the signer to extract a set of common translation rules. Additional rules were also compiled from the BDSL language development book. Then we used the annotated data and the compiled rule set to perform data augmentation techniques like Rule-based Morphological Transformation, Masked-Token Substitution and Retrieval-Augmented Generation (RAG) based Few-Shot Prompting to generate 3000 more sentence-gloss pairs, resulting in a dataset of 4000 text-gloss pairs. Finally we finetuned some state-of-the-art multilingual transformer models on our datasets to observe its performance. The models include mBART-50, mT5, NLLB, GPT-4.1-nano. We then evaluated the models performance based on the two key metrics - BLEU score and Comet score.

3.1 Workflow

a) Dataset Creation

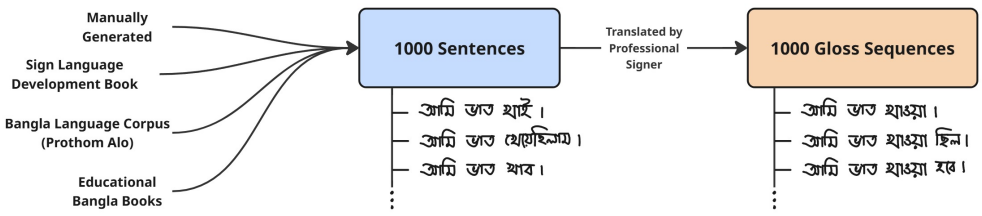


Figure 3.1: Workflow for Dataset Creation

b) Compiling Rule Set

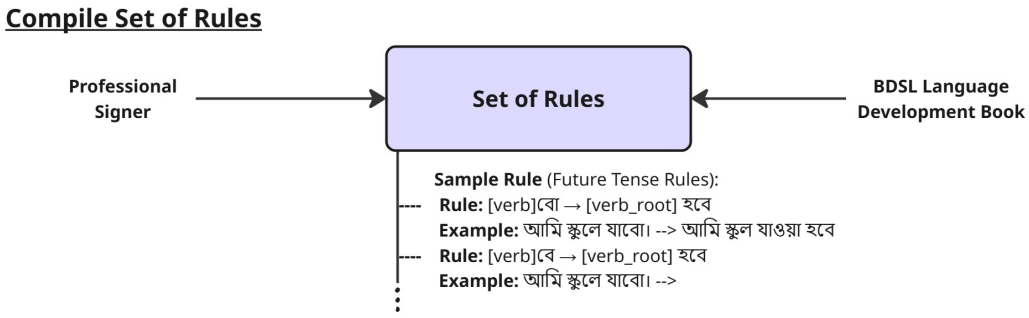


Figure 3.2: Methodology for Rule Set Compilation

c) Data Augmentation

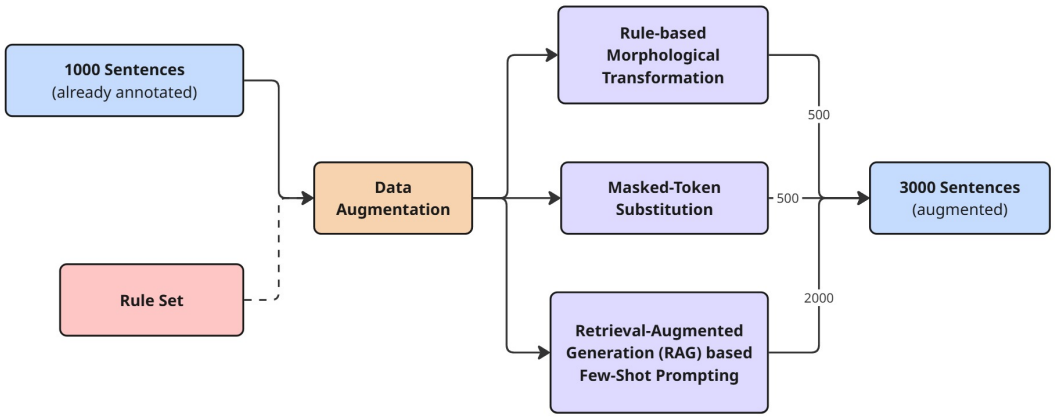


Figure 3.3: Data Augmentation Process

d) Model Finetuning and Evaluation

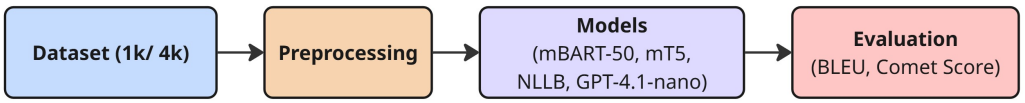


Figure 3.4: Model Finetuning and Evaluation Workflow

3.2 Dataset Collection

Bangla sign sentence and gloss pairs remain significantly underrepresented in publicly available datasets. So, we took inspiration from the widely used RWTH-PHOENIX-Weather 2014T dataset for German Sign Language. Our initial plan was to extract Bangla sentences from televised news segments and have them annotated to gloss sequences by a professional signer. Therefore we requested Bangladesh Television for access to their new segments that included sign language interpretation. Afterwards, we were provided with 10 hours of news videos and were referred to multiple signers. We reached out to them and after a few meetings we decided to hire Mr. Ariful Islam, an experienced Bangla sign language presenter that worked there and also a professional sign language interpreter and trainer as our gloss sequence annotator. From our discussions with him we understood that translating isolated Bangla sentences with specific sentence structures and themes will yield better results. We also visited Ramna Intellectually Disabled and Autistic School to get a better grasp on how sentence construction in Bangla sign language is taught from a preliminary level. From here, we learned that there are no specific Bangla Sign language books that are followed while teaching, rather teachers would train from their experience. However, we came across a textbook from DeafChild worldwide for Bengali sign language through their official website. This book gave a clear representation of how Bangla words in a sentence would morph when they were converted into a gloss sequence. So we first created a batch of 1000 Bangla sentences on different themes and then had our professional signer annotate it accordingly. In addition to this when forming these Bangla sentences we assigned many features of the sentences as the metadata for an individual sentence to help with the augmentation process.

3.3 Dataset Description

Our dataset consists of 4000 Bangla sentences which are paired with their corresponding gloss translations. The Bangla sentences were created based on a particular theme as it allowed sentences to have a definite subject or information to convey, similar to how in sign language, signs are often structured around a clear subject. By taking this factor into consideration the annotated gloss sequences provide a more consistent representation of how actual signs are performed. We have chosen sentences from 19 different theme categories to maintain variation and reduce bias in our dataset.

The themes include Crime, Art , war , asking for information, time , weather, education , Family and Relationships, Bangladesh Seasons, Festivals , Accident , Holiday, Tourism, Health and disease, Animal, market and food, natural disasters, Technology, sports. Additionally we have also selected sentences from multiple news-related open-source Bangla corpuses. This is done to capture more formal and domain-specific sentence structures that are used in a more public setting. We also annotated the type of our sentences with simple, compound and complex and subsequently the tense with simple present, present continuous, present perfect continuous, and their respective forms in the past and future tenses. Features like tense and sentence type are pivotal in the data augmentation steps because they

Field	Value
Sentence	আমি আমাদের পরিবারের সবার ছোট।
Gloss	আমার পরিবার ভাই-বোন ভিতর আমি ছোট।
Theme	Family
Sentence_Type	Simple
Tense	Present
Gen_Type	Human_annotation

Table 3.1: Metadata for a single Bangla sentence-gloss pair

enable controlled generation of syntactically and temporally diverse sentences and equivalent gloss sequences derived from grammatical rules. To keep track of the synthetic, rule-based augmented data and the manual annotated data we use Human_annotation, RAG_AUG, TENSE_AUG, CONTEXT_AUG to differentiate. By organizing in this format our dataset will open more scopes in both Bangla sign language and Natural Language Processing research. We have compiled our dataset into a csv format file as shown in table 3.1.

3.4 Data Preprocessing

Preprocessing plays a pivotal role in NLP tasks in transforming unstructured raw text into a cleaner and ordered format. This is especially true for a complex language like Bangla, and therefore, to improve the model's performance, careful and thorough preprocessing becomes even more critical.

3.4.1 Unicode Normalization

In Unicode, a character can be represented in multiple ways such as a composed form or a decomposed form depending on how base characters and diacritics are stored and processed. So, the tokenizer may create separate tokens for the same word, which may lead to inconsistencies and inefficiencies during fine-tuning. For example, for the Bangla word: 'মিষ্টি' the composed form may be tokenized as: ['মিষ্টি'] and the decomposed form as: ['মি', 'ষ', 'ি্টি']. Thus, we performed Unicode normalization on the Bengali sentences and their corresponding glosses to ensure consistency in the tokenization process.

3.4.2 Whitespace & Special Character Trimming

We have removed extra white spaces from the text and gloss blocks of our dataset as these extra spaces between words may lead to the creation of new unwanted tokens. Furthermore, while we have decided to keep punctuations such as '?', '|', '!' as they express the tone of the sentence and also signify the end of a sentence, we have also made sure that unnecessary symbols such as '@', '#', '$' etc are removed from the dataset as they are redundant for sign language and if left may cause unnecessary token splits.

3.4.3 Tokenization

Tokenization is the process of splitting a text into smaller units such as words, subwords, or characters, which are also known as tokens. Machine Learning and Deep learning models can both leverage the use of these tokens to perform NLP tasks. In this paper, we have applied SentencePiece tokenization to tokenize our dataset before feeding it into transformer-based models. SentencePiece works by processing the input as a continuous stream of characters, which aligns well with Bangla, which does not necessarily always use whitespace. We extracted both the sentence and gloss sentences from the dataset and tokenized them with a maximum sequence length of 128 tokens.

3.4.4 Truncation & Padding

We set Truncation to True and padding to the maximum sequence length. Truncation limits any sequence by its maximum length, which we set to 128 by removing excess tokens from the end. Padding does the opposite, that is it adds extra special padding tokens to ensure all sequences in a batch have equal length. Together, they play a key role in preprocessing, where they ensure both input and output sequences have uniform length. This is essential for batch processing and thus for ensuring efficient computation.

3.4.5 Train-Validation-Test Partitioning

We performed an 80/10/10 train/validation/test split on our original dataset of 1000 expert-annotated sentence-gloss pairs. The same test set was used for testing the augmented dataset to ensure fair evaluation.

3.5 Preliminary Data Analysis

3.5.1 Dataset Overview

The final augmented dataset comprises 4,000 text-gloss pairs, representing a four-fold increase from the original 1,000 professionally annotated pairs. The dataset distribution shows RAG-based augmentation contributing 53.0%, human annotation 24.9%, morphological augmentation 12.5%, and contextual augmentation 9.6%.

3.5.2 Length Analysis

In our dataset, the sentences average at 44.5 characters and 7.0 words per sentence, while BDSL gloss averages 45.7 characters and 8.1 words. The distributions show that both text and gloss follow normal patterns, with text peaking around 40–50 characters and gloss around 45–55 characters. At the word level, text shows a concentrated distribution around 6–8 words, while gloss exhibits a broader distribution centered around 7–9 words. This indicates the structural transformations required in sign language translation.

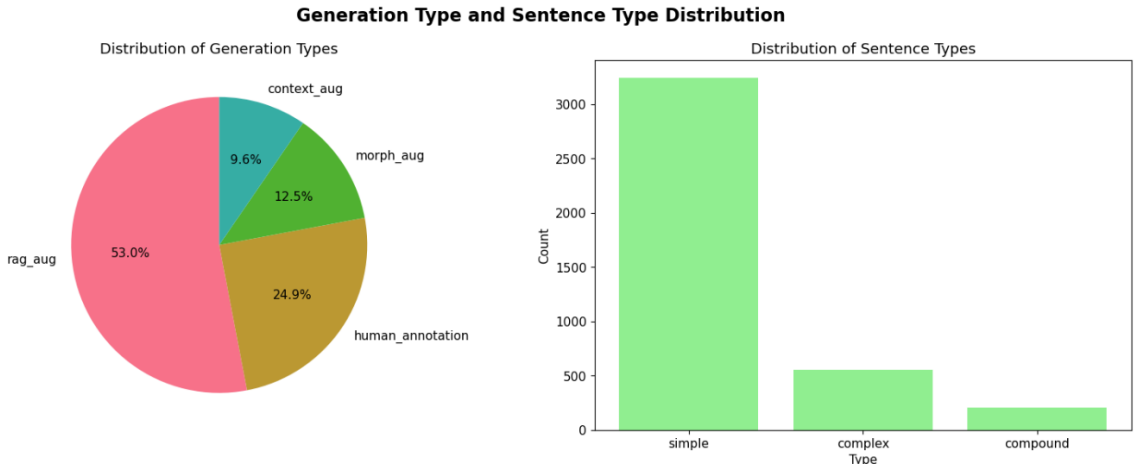


Figure 3.5: Distribution of data by generation type (RAG, human, morphological, contextual)

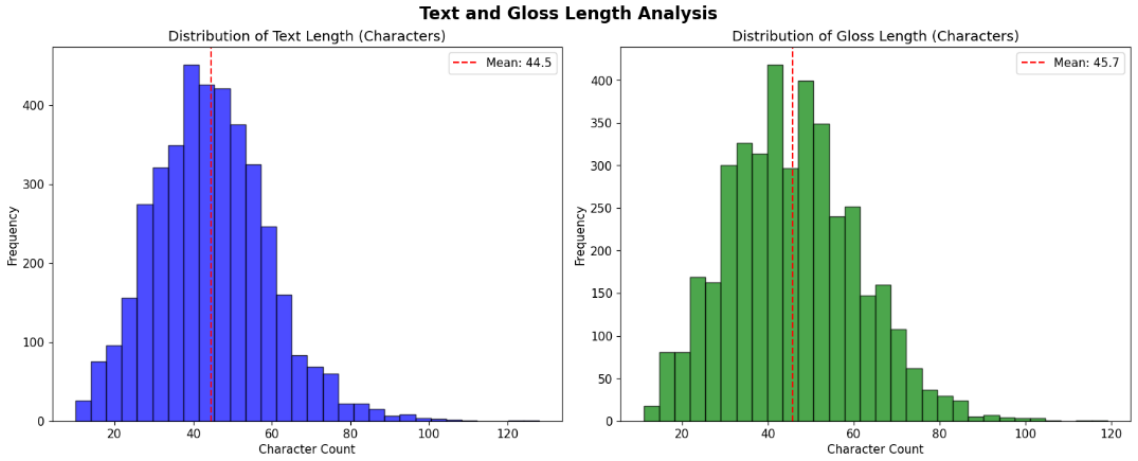


Figure 3.6: Length analysis of Bangla text and BDSL gloss in characters

3.5.3 Tense Distribution

As for tense distrubution, simple present dominates the dataset with 1,477 instances (36.9%), followed by simple past (843 instances, 21.1%) and simple future (566 instances, 14.1%). Present continuous accounts for 441 instances (11.0%) and present perfect for 392 instances (9.8%). Complex tenses such as present perfect continuous (16 instances) and future continuous (10 instances) appear minimally.

3.5.4 Topic Coverage

The dataset spans 23 distinct topics, with food being most prevalent (507 instances, 12.7%), followed by family and relationships (473 instances, 11.8%), and crime (413 instances, 10.3%). Other significant topics include sports (392 instances, 9.8%) and education (374 instances, 9.4%). The distribution reflects real-world communication patterns, with everyday subjects like art (272 instances), health and disease (226 instances), and technology (159 instances) being well-represented. Less frequent topics include specialized domains like transportation (12 instances) and environment (7 instances).

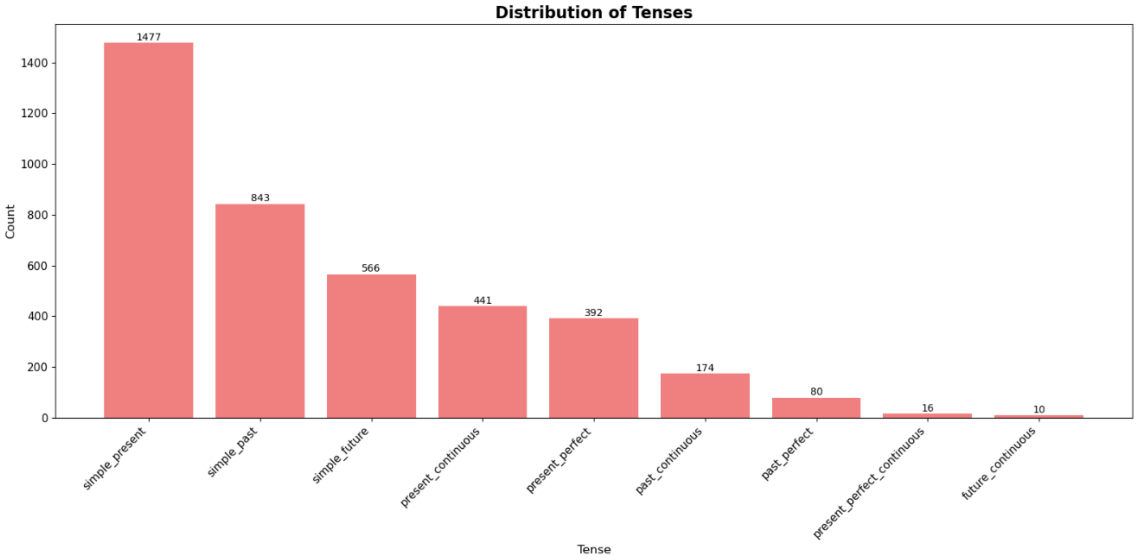


Figure 3.7: Tense distribution in the dataset

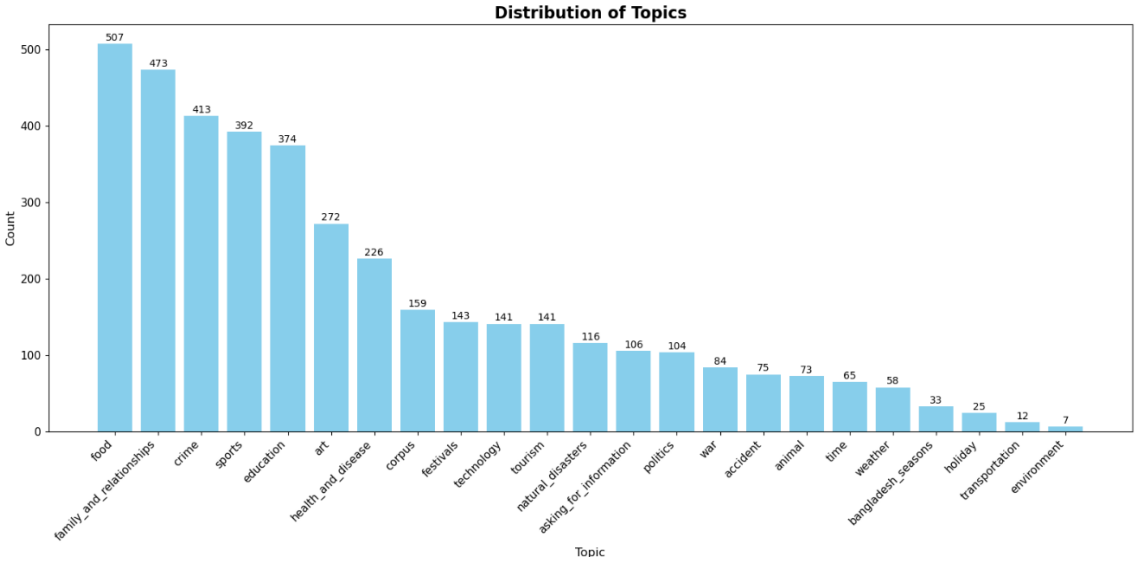


Figure 3.8: Distribution of topics covered in the dataset

3.5.5 Sentence Structure

Simple sentences comprise the vast majority of the dataset with approximately 3,200 instances (80%), followed by complex sentences at around 550 instances (13.8%) and compound sentences at approximately 250 instances (6.2%). This distribution aligns with sign language preferences for clear, comprehensible structures and reflects natural communication patterns where simple sentence constructions facilitate better understanding and translation accuracy.

3.6 Data Augmentation

Data augmentation plays a crucial role in addressing data scarcity, especially in low-resource language tasks where annotated data is limited. Specifically in our case of Bangla Text-to-BDSL gloss translation, we curated a dataset of 1000 sentence-

gloss pairs annotated by a professional signer. While this dataset is larger than other similar works in the field of Bangla text to gloss translation, it still remains insufficient in achieving satisfactory results in model training. As such, we applied several data augmentation strategies to enhance the dataset which include: (1) Rule-based Morphological Transformation, (2) Masked-Token Substitution, (3) Retrieval-Augmented Generation (RAG). With these methods, we generated a total of 3000 new entries of text-gloss translations. Our final dataset now has 4000 entries with 1:3 ratio for original to augmentation.

3.6.1 Rule-based Morphological Transformation

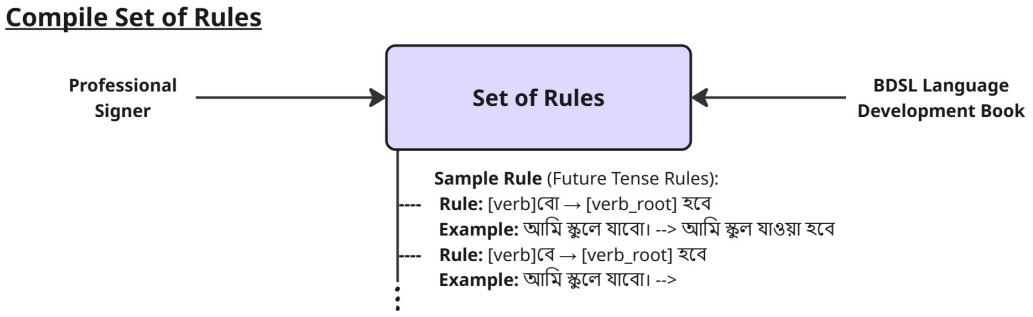


Figure 3.9: Compiling Set of Rules

We collaborated closely with a professional signer to analyze and extract common translation patterns from our annotated dataset. A significant portion of these patterns were related to tense. Each tense follows some consistent morphological rules that maps the verb of the text form into gloss. For example a common pattern in future tense is that the verb is translated to [verb root] + "হবে". For instance, "আমি কাজ করব" gets translated to "আমি কাজ করা হবে". Based on such consistent mappings, we, along with the help of the professional signer, compiled a set of rules. Afterwards, we applied these rules to augment existing sentences by systematically converting them into other tenses. For example, the sentence "আমি ভাত খাই" was transformed into its past and future tense forms, such as "আমি ভাত খেয়েছিলাম" and "আমি ভাত খাবো" , respectively. This method ensured that the augmented sentence-gloss pairs remained linguistically accurate and semantically faithful to the original meaning. Furthermore, because the glosses were generated using verified rules rather than arbitrary transformations, the resulting data maintained a high level of consistency. With this, we manually generated 500 text-gloss pairs and expanded the dataset in a controlled and interpretable manner, which is particularly important in low-resource tasks such as Bangla text-to-gloss translation.

3.6.2 Masked-Token Substitution

The second method we applied is a form of Contextual Augmentation known as Masked Token Substitution. This is used to create lexical variants of sentence-gloss

pairs using known rules and with the help of Bangla-Bert. Firstly, we define templates like "<mask>টি ভাত খাচ্ছে", where certain words (typically common nouns or verbs) are masked. Then we run it by Bangla-Bert which predicts the most contextually appropriate replacements for the masked tokens (<mask>). The newly generated sentence preserves the grammatical rule and structure of the original sentence. This approach was particularly motivated by issues observed during preliminary model running such as - the model frequently failing to generalize certain transformation rules like converting "ছেলেটি" to "ছেলে এটি". Despite being a consistent rule in our dataset, the model struggled to apply it across similar variations. As such, by generating diverse sentence forms through this method, we aimed to expose the model to a broader range of patterns, improving its generalization capability, while maintaining the rules of text-to-gloss translation. We created 500 text-gloss pairs with this method.

3.6.3 Retrieval-Augmented Generation (RAG) based Few-Shot Prompting

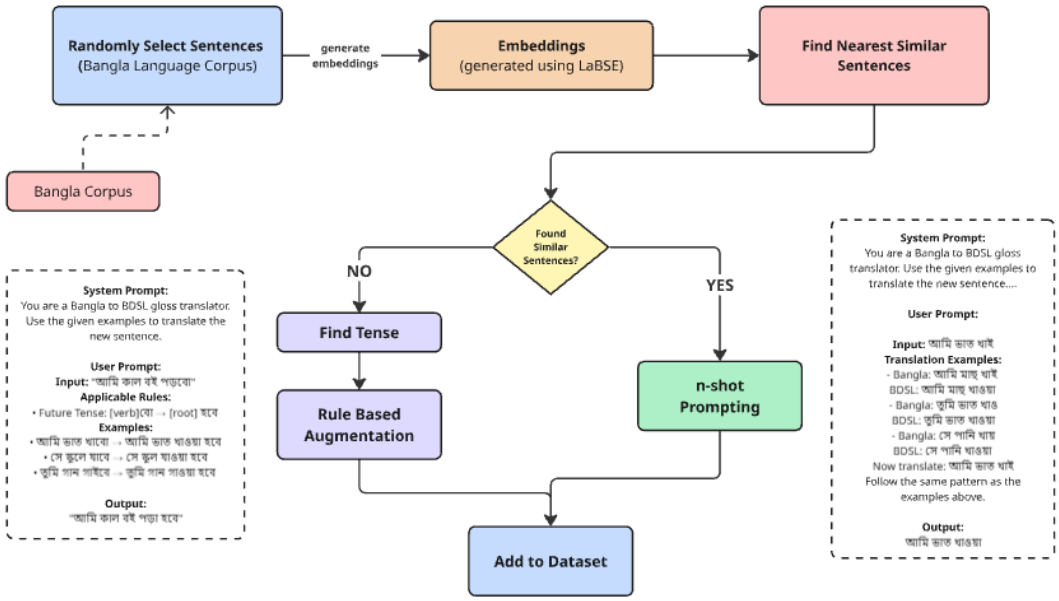


Figure 3.10: Methodology of Retrieval-Augmented Generation (RAG) based Few-Shot Prompting used for data augmentation

We used LaBSE embeddings and Pinecone as the vector database to retrieve similar sentence-gloss pairs, and employed GPT-4.1-mini for few-shot prompting and chain-of-thought-based gloss generation. Retrieval-Augmented Generation (RAG), paired with few-shot prompting, is particularly effective for data augmentation in low-resource settings because it grounds a model's generation by providing contextually similar examples to learn from. Particularly in our case of Bangla text-to-gloss translation, it enables the model to produce more accurate translations even for some rare or less frequent sentence structures that are not covered by the common rule sets we compiled. For example, some words do not have direct translations in BDSL.

For instance, the word রেসিপি ("recipe") does not have a corresponding sign and is instead translated to খাবার রান্না নিয়ম . Without giving such context during data augmentation using GPT, these patterns may not show up in the output. As such, we implemented a RAG based approach that first retrieves the similar occurrences from our professionally annotated dataset. This way, if an input sentence contains a word like রেসিপি , the system is likely to retrieve examples containing similar or same terms. These retrieved examples, along with their corresponding glosses, are then included in a prompt passed as a few shot prompt. The model, having access to these grounded examples, is better guided to produce an accurate and contextually appropriate gloss for the new sentence. We set a similarity score threshold of 0.5, meaning that if the score is above 0.5, that is considered a similar sentence usable for few shot prompt. To prevent overload, we also cap the number of retrieved examples at 20 per prompt. Now the problem arises if less or no similar sentences are found. To address this, we implement a fallback mechanism where, instead of sending n similar sentences, it sends the rules in the prompt. We further improved this approach by implementing a two-stage prompting strategy. Instead of sending all the rules at once, which may overwhelm the model due to the complexity and volume of the input, we first issue a prompt to identify the tense of the sentence. Then, based on the tense, we only send the rules of that particular tense along with the examples. This way the prompt is more focused and likely to generate better results. This two-stage prompting strategy draws on principles similar to chain-of-thought prompting, where an intermediate reasoning step (tense identification in our case) is used to refine the final generation prompt. We generated 2000 text-gloss pairs using this approach, and applied the Cohen's kappa method to validate our results.

3.6.4 Validation using Cohen's kappa

Cohen's kappa is a widely used statistical measure for evaluating how consistently different annotators label the same data. It is particularly useful in validating data augmentation techniques in low-resource settings where ground truth data is limited. In our case, we first generated 1000 text-gloss pairs using our RAG based approach. Then, we randomly selected 15% of the text-gloss pairs and asked two professional BDSL signers to independently review and validate the gloss translations. The signers who validated these translations are Mr. Ariful Islam and Mrs. Tanjila Tartushi, both of whom are professional sign presenters at BTV. They were given 2 input fields to validate each entry. The first field was a Yes/ No question asking if they think the gloss is understandable. The second field asked them to rate the gloss on a scale of 1-5. We then calculated the Cohen's kappa score between the two signers' judgments. Our hypothesis was that a kappa score above 0.60 would be considered a minimum acceptable threshold. The results showed a kappa score of 0.7489 which means substantial agreement. As such, we concluded that our data augmentation process was valid. A detailed overview of our findings is presented in the results section.

Chapter 4

Experiment Setup

This following work investigates the leveraging of three separate Transformer-based models, mBART, NLLB-200, as well as mT5 for translating Bengali-to-Bangla Sign Language glosses, centred around a low-resource task. The following models all consist of an encoder-decoder architecture and are pretrained with multiple language corpora, ensuring their suitability in low-resource data cases. mBART is a multilingual extension of BART, focusing on their denoising ability to create embeddings bearing contexts across multiple languages and fine-tuned to create corresponding gloss output from Bangla texts. NLLB-200, on the other hand, is a huge multilingual Transformer model using Mixture-of -Experts layers and curriculum learning as well as dynamic sampling to enhance high quality performance on low-resource linguistic data like Bangla. Its powerful system makes it muster up efficient transfer learning and thus generate intended gloss sequences from their corresponding Bangla sentences. Finally, mT5's strong test-to-text pipeline, where it frames every task as a text input and generates a text output, along with its multilingual pretraining on the mC4 corpus, makes it suitable for Bangla sentence-to-gloss conversion.

4.1 Model Description

4.1.1 Understanding the mT5 Model for Multilingual Tasks

Google's mT5 [15] is another sequence-to-sequence Transformer model that is trained on an extensive dataset called mC4 that covers 101 languages, including Bangla. It follows T5's [7] architecture. The architecture is very similar to a standard transformer model. Firstly, input sequences are tokenized and mapped to vector embeddings, which are fed to the encoder. The encoder is made up of different layers with each having two subcomponents which are the self-attention layer and a small feed-forward network. A simplified layer normalization scheme and a residual skip connection is applied to each subcomponent. Dropout is applied to the feed-forward network, skip connection, attention weights, and input and output of all layers. The decoder is equipped with a self-attention mechanism that enables the model to focus on past outputs. The output from the final decoder block is passed to a dense layer with a softmax output. MT5 also uses a unique positional embedding scheme where instead of learning a fixed position value, it learns how far the next token actually is from the current position. Only 32 embeddings are used for memory efficiency. This architecture enables its special "Text-to-Text Transfer

Transformer" label. This Text-to-Text Transfer refers to the fact that mT5/T5 are specifically trained to cast any form of NLP tasks as text input and generate some text output, thus allowing for a multitude of tasks to be performed by the same model with the same hyperparameters. Sentence-to-gloss translation is a text-to-text task that mT5 excels at. Furthermore, mT5 is already pretrained on the Bengali language and is thus knowledgeable on Bengali sentence structures and nuances. So this knowledge can be leveraged to better aid a low-resource task like sign language sentence to gloss conversion. Taking into account all these factors, along with its solid performance when finetuned for downstream tasks, mT5 inherently feels like a great option for our specific task.

4.1.2 Text-to-Gloss Conversion using mBART

Liu et al.[10] explored the multilingual form of the denoising model, mBART constituting pre-training intended at leveraging the translation capability via machines, encompassing diverse languages with the aid of diverse one-language datasets. This model implements various approaches like permutations on sentences and masking on word-span to alleviate training quality, thus leading to better output compared to other contemporary machine translation methods under supervised as well as unsupervised circumstances. The results suggest that mBART yields considerable alleviation in performance, particularly in the domain of low resource or medium resource environments for translation in the machine-level. The outputs constitute coherent word-flow structure and hold significant accuracy to the exact output values. In addition, the capability of mBART to draw generalization among languages signifies its diversability and extensive prospects in the machine translation domain.

mBART is a transformer model architecture which constitutes a Seq2Seq approach intended for translation via machine and other text generation objectives including high prospects in sign language conversions in multiple languages. It is an extension of BART (Bidirectional and Auto-Regressive Transformer) which is limited due to its monolingualism and thus offering limited resources. On the other hand, mBART has been trained in multiple differing languages which includes many low-resource languages i.e. Bangla being one of them. Hence this leveraging architecture offers a promising prospect for converting Bengali language text to corresponding language glosses which is a vital component of our exploration.

The architecture of mBART relies on two distinct separated phases constituting the encoder which processes the whole input sequence (i.e. Bengali sentences) at once bi-directionally and then converts them into embeddings bearing contexts and the decoder which generates the output gloss sequence in a sequential step-by-step manner. The encoder generates multilingual embeddings which enable the transfer of knowledge among multiple languages. mBART is pre-trained with the aid of a denoising autoencoder objective which enables reconstruction of faltered input sequences. This influences the capability to generate accurate and relevant output sequences even during dealing with inputs containing noises, thus enhancing its own effectiveness in this particular domain of Bangla text-to-corresponding gloss conversion.

The task goes as:

Let X represent the Bengali sentences, i.e.,

$$X = \{x_1, x_2, x_3, \ldots, x_N\} \tag{4.1}$$

Let Y represent the gloss sequences, i.e.,

$$Y = \{y_1, y_2, y_3, \ldots, y_N\} \tag{4.2}$$

where N represents the length of the source sequence and T represents the length of the target sequence.
The encoder processes the input X, the Bengali sentences to produce embeddings with contexts, denoted via H.

$$H = \text{Encoder}(X), \quad H \in \mathbb{R}^{N \times d} \tag{4.3}$$

where d is the concealed dimension of the model.
The decoder generates gloss sequence Y, one token-after-another sequentially:

$$P(y_t | y_{<t}, H) = \text{softmax}(W z_t + b) \tag{4.4}$$

where:

- z_t is the embedding of the decoder output at timestamp t,
- W, b are the projection weights and bias for the word meanings and relevance.

$$X \to \{i_1, i_2, \ldots, i_N\}, \quad Y \to \{j_1, j_2, \ldots, j_T\} \tag{4.5}$$

Uniformity is ensured by padding shorter sequences, and longer sequences are constrained within a maximum limit. Fine-tuning is done by minimizing the cross-entropy loss function.

At each timestamp, the model evaluates a probability over the vocabulary, P_t:

$$P_t = [P_t(w_1), P_t(w_2), P_t(w_3) \ldots, P_t(w_V)] \tag{4.6}$$

where V represents the vocabulary size. The true token is shown via a structure:

$$y_t = [1, 1, 0 \ldots 0, 1] \tag{4.7}$$

$$L_t = -\log P_t(y_t) \tag{4.8}$$

where L_t represents the cross-entropy loss.

$$\theta \leftarrow \theta - \eta \nabla_\theta L \tag{4.9}$$

where θ represents the model parameters and η is the learning rate.

The model runs iterations through several epochs as the loss gets smaller.
Once fine-tuning is completed, inference is carried out as the encoder creates embeddings bearing contexts H from X.
The decoder generates gloss tokens Y_{pred} step-by-step:

$$y_t = \arg\max P(y_t|y_{<t}, H) \tag{4.10}$$

The Greedy decoding constitutes the selection of the greatest probability at each step, while beam search constitutes the exploration of multiple token sequences in a parallel manner and picking the best.

Strength of mBART in this ground:

As discussed earlier, mBART constitutes multilingual operations by pretraining through differing multilingual elements with language-defined tokens. It comprises generating multilingual embeddings which enables the transfer of knowledge among high-resource to low-resource languages such as Bengali and thus offers huge prospects for conversion of Bengali text to corresponding sign language gloss. The important tag $< bn\_IN >$ enables the model to utilize the shared characteristics among related languages while also preserving language distinction.

Hardware limitations and others:

The mBART models are huge, i.e., having an extensive number of parameters, thus requiring imposing computational power for training as well as inference. Without high computational power, the process of fine-tuning and inference will be extremely slow. It is necessary to use a GPU with considerable memory size (over 16 GB) to prevent memory flooding when fine-tuning takes place. Transformers tend to scale in a quadratic manner with the length of the sequence, and thus Bengali sentences with longer lengths will add extra complexity resulting in higher delay during fine-tuning and inference.

Glosses use a special structure and context which cannot be grasped by mBART's pre-training. High-quality datasets are required, and while mBART makes use of large multilingual corpora, a Bengali-to-gloss dataset can still cause overfitting. Glosses have a differing set of semantics and word flow and sequence compared to natural spoken language, which emphasizes fluency and meaningful flow structure. mBART mainly focuses on generating that by creating unnecessary fillers within the output sequence which deviates from the intended fragmented gloss representations.

This problem can, however, be compensated by systematic filtering to get rid of unnecessary fillers to effectively resemble the intended sequence of gloss. In order to compensate for hardware-based problems, mixed precision training could be employed, which involves both 16-bit and 32-bit floating-point executions while training. Sensitive portions are done via fp32 for stability while other computations are changed to fp16 to reduce memory usage. Fragmenting the large training dataset across multiple processing units enables parallel simultaneous processing, and PyTorch's DDP is well-known for well-spread training maintained with synchronicity.

4.1.3 Text-to-Gloss Conversion using NLLB

NLLB Team et al.[16] ventured heavily upon this robust model which constitutes a highly adaptable translation approach which focuses on multiple languages. Therefore it has the ability to translate more than 200 languages along with a language pair count of thousands, which also includes many low resource languages.

The core architecture comprises a translation mechanism which prevents the overfitting issues of a robust model while being trained with a low resource, also it evades the major issue of low resource getting completely neglected in the presence of high resource dataset. This model, in fact uses an approach differing from traditional Transformers which uses layer normalization once residual connection is done i.e. after each sublayer which is known post-LN. However multiple layers are placed after one another, in case of deeper models, the approach of post-LN becomes vulnerable to delayed convergence as well as unstable tendency of the gradient. In the method described in this paper, the normalization of the layers is done before the input data enter the sub-layer's major processing systems including self-attention and cross-attention as well as feedforward. Hence the following sublayer is being trained with the aid of an input that is already normalized, thus increasing the rate of convergence as well as enhanced stability of gradient. NLLB uses a Mixture-of-Experts layer where the regular sub-layer, namely the feed-forward, is substituted by another sub-layer of MoE. Therefore the pre-existing layer is displaced by 128 experts in a single layer - and each of these experts are individual FFNs, constituting their own collection of parameters. However, among these 128 experts, not all of them are used at once for one token, rather it chooses top-k where k=2. This process is referred to as sparse expert activation, and this process is computationally similar to the pre-existing FFN as for one token only 2 experts are utilised.

The main benefit of MoE is that its use of 128 separate experts allows each expert to focus more on differing linguistic patterns of different language families thus enabling efficient translation, particularly in a multilingual environment. Since the model itself is huge(54B parameters), for any low resource data, it is very likely to overfit via memorization, and also cause frequent use of certain experts leaving the others idle. To avoid this, there are few regularization techniques which include dropping out of some outputs based on some randomly selected probability value, for example 30% of the output could be set to zero. This method however does not solve the problem of the same expert being used repeatedly and also it increases the number of Floating-Point operations causing rise in computational cost. Another important method that could be used here is EOM which suggests choosing two particular experts, and applying masking on one of them randomly. Therefore this approach prevents excessive reliance on any particular expert, also enhancing significantly better performance for low resource datasets. Another approach known as CMR, adds a trained gateway illustarted in the following expression(4.11):

$$g(x_t) \in [0, 1] \tag{4.11}$$

which can choose between a conventional FFN and MoE sublayer constituting experts demonstrated in the equation(4.12):

$$\mathrm{CMR}(x_t) = (1 - g(x_t)) \cdot \mathrm{FFN}_{\mathrm{shared}}(x_t) + g(x_t) \cdot \mathrm{MoE}(x_t) \tag{4.12}$$

Hence it implies that in some cases, tokens associated with low resource do not have to go through heavy MoE procedure, rather they could depend on the regular lower cost FFN. Therefore a loss term is incorporated in here signifying the utilisation of MoE within a limiting value. This is shown in the expression(4.13):

$$L_{\mathrm{CMR}} = \frac{1}{T} \sum_{t=1}^{T} |g(x_t) - p| \tag{4.13}$$

where

- p = target proportion of MoE usage,
- $g(x_t)$ = learned gating value for token x_t,
- T = total number of tokens.

Despite the ability of this technique to choose between MoE as well as early FFN in a dynamic manner, the findings suggest that this method either equaled or performed poorly compared to the EOM. This has also yielded an increased FLOPs count compared to EOM by around 23%.

Another approach taken by this model for balancing data is known as Dynamic Example Strategy which focuses on low-resource linguistic data to a greater extent compared to high resource linguistic data. This prevents the model from turning out to be heavily skewed at languages with greater amounts of available data. The next method is known as curriculum schedule which constitutes an organised scheme where the model is at first trained with the high-resource data and the model undergoes fast convergence. Once it gains strong foundational knowledge on language and translation illustrations, the scheme gradually focuses towards lower resource language models.

In order to train a huge multilingual model with being dependent on insufficient bitext annotated by humans, NLLB-200 focus on three major techniques: first one being Bitext mining which uses LASER3, finding semantically close pairs of sentences through a monolingual corpora to produce parallel data which are synthetic. The next technique is Back-translation in which data associated with the intended language is translated to the source language with the aid of a model which has been pretrained thus generating simulated pairs of synthetic sentences which are not translated by humans. The third technique is where NLLB-200 is trained via generating corrupt inputs. However this method was comparatively less helpful for NLLB-200.

Although NLLB-200, is inherently designed for translating text from one language to another, it can also be used for translating a low resource language like Bengali to Sign Language Glosses in Bangla(BSL). In this case, the BSL itself is thought of as a different intended language. The new language tag is defined as e.g., `<lang_tgt=BSL_GLOSS>` and a simultaneous corpus containing Bengali sentences is created. The model can be fine-tuned with the aid of pretrained encoders in Bengali and multilingual prospects. Due to the robustness of the NLLB-200 model along with its MoE, it can generalize in a decent manner even with limited data. However this method enables efficient maneuvering of the mechanism of the model to be able to perform this task but this also demands meticulous planning of the datasets.

4.2 Fine tuning Setup

4.2.1 Fine-tuning mT5 for Bangla Gloss Generation

According to Xue et al[15] the mT5-small was pre-trained on a new Common Crawl-based dataset covering 101 languages and has 300M parameters. Therefore the model already understands the underlying structure behind Bangla words and sentences, allowing us to specialize it in our Bangla sentence to gloss generation tasks. We load our Bangla-sentence and gloss pairs with a batch size of 16 because higher batch sizes like 32 and 64 would require a lot of memory for computation. We set our learning rate to 0.001 with warm-up steps set to 50 allowing the model to gradually linearly increase its learning rate to 0.001, preventing large initial updates to the weights and biases which can destabilize the model. Epoch was selected at 20 with early stopping implemented, to prevent overfitting and ensure optimal model selection. Additionally, we also used AdamW optimizer for better regularization during training.

4.2.2 mBart-50 Fine-tuning Parameters

The mBart-50 model was pretrained on a diverse set of 50 languages which included Bengali and was primarily developed for fine tuning on translations tasks [11]. Based on this the model was fine tuned on the following hyper-parameters. An effective batch size of 16 was selected with the learning rate set to max 3e-5 with warm-up steps set to 300. An epoch of 20 was selected with early stopping implemented.

4.2.3 Parameter-efficient Fine-tuning of NLLB-200

NLLB-200 was primarily optimized for machine translation tasks. There are three variants of this model: NLLB-200-distilled-600M, NLLB-200-distilled-1.3B and NLLB-200-3.3B. The models are pre trained on a massive multilingual corpus making them eligible for the Bangla sentence to gloss translation task. We evaluated the performance of the NLLB-200-distilled-1.3B model to assess how a billion-parameter architecture would perform on our dataset. However, conducting a full parameter finetuning of a 1.3B model would require significant computational resources. Therefore, we applied Parameter-Efficient Fine-Tuning (PEFT) techniques, specifically Low-Rank Adaptation (LoRA) which enables effective model adaptation while maintaining computational efficiency [12]. LoRa provides extra trainable parameters which are known as low rank matrices, while preserving the original pre-trained weights thus the original pre-trained knowledge is less modified. Due to our dataset being relatively small we set the rank parameter (r) to 8, which determines the dimensionality of the low-rank matrices and the LoRA. The alpha parameter was set to 16 for a scaling factor 2 so that the learned weight updates are doubled before being added to the original weights, thus allowing the pre-trained model to learn task specific patterns. A dropout rate of 0.05 was applied specifically to LoRA layers to provide regularization and prevent overfitting of the adapter weights. The target module section included q_proj, k_proj, v_proj, out_proj of the self-attention layers, and fc1, fc2 in MLPs. This allows the model to adapt in terms of both context understanding and feature transformation for gloss sequence generation. The batch size was set to 1 with gradient accumulation steps of 24, creating an effective

batch size of 24. Gradient accumulation maintains training stability and enables effective parameter updates despite the small per-device batch size. These steps were mainly taken due to memory limitations, as larger batch sizes would exceed available VRAM. The learning rate was set to 3e-4 with the training epochs set to 5 and mixed precision training (FP16) was enabled to reduce memory usage and accelerate training while maintaining numerical stability. Despite these efforts, we were only able to run the model on the manual annotated part of our dataset.

4.3 Evaluation Metrics

For model evaluation we used two metrics COMET and BLEU. We used BLEU (Bilingual Evaluation Understudy) scores to evaluate the quality of the text to gloss translations. BLEU is a precision-based metric that measures the overlap between the predicted gloss and the reference gloss at the n-gram level. BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores, respectively corresponds to unigram, bigram, trigram, and 4-gram matches. BLEU-1 focuses on word-level accuracy, while higher-order BLEU scores penalize lack of fluency and structural consistency by requiring longer contiguous matching sequences. BLEU is limited in capturing semantic similarity but it provides a measure of lexical overlap. COMET evaluation uses a pre-trained cross-lingual encoder model to extract contextual embeddings of the predicted and reference gloss sequence along with the source Bangla sentence. Afterwards it uses layer-wise attention pooling to get sentence-level embeddings. It then computes the products and absolute differences between the predicted gloss sequence and the source and reference embeddings then these values are concatenated with the predicted and reference gloss sequences embeddings. This helps the model understand how similar or different the prediction is compared to the source and reference based on semantic similarity and contextual relevance. The combined features are fed into a simple neural network that predicts a quality score. Therefore by capturing the semantic similarity and the lexical overlaps these metric can help in understanding the quality of our predicted Bangla gloss sequence against both the original Bangla sentence and the reference gloss sequence.

Chapter 5

Results and analysis

5.1 Loss Comparison Graphs for the fine-tuned models

The Figure 5.1 (left) shows the loss curve of training for the mT5 model with the dataset having a count 1000 (i.e. 1K sentence pairs) and the Figure 5.1 (right) shows the the loss curve of training for the mT5 model with the dataset having a count 4000 (i.e. 4K sentence pairs). Both of the two curves show a regular fall in the loss throughout the course of training along the total epoch up until 20, this conveys the proper learning behaviour of the model and its ability to optimize. The fine-tuning behaviour is reflected in the significant fall in loss within 0 to 5 epochs, as they both approach convergence. For the 1K dataset, loss becomes steady from epoch 10 i.e. starts flattening, reaching a loss value in between 1.5 to 1 while for 4K dataset, loss becomes steady from epoch, reaching a loss value of 1. Therefore it shows a stronger better generalization capacity in case of a dataset with higher count.

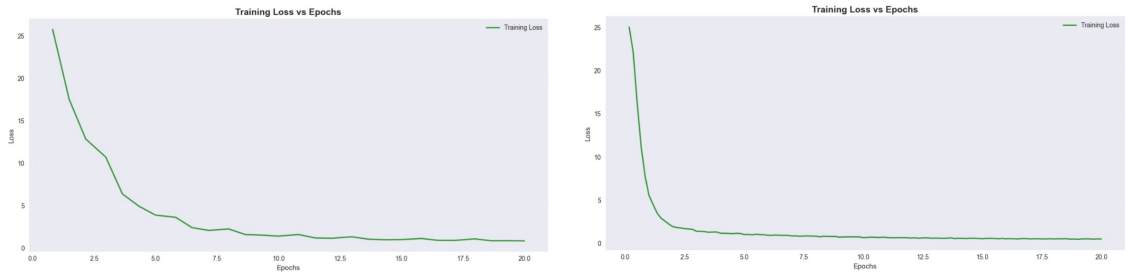


Figure 5.1: Loss convergence for mT5: left = 1000 sentence-gloss pair dataset, right = 4000 sentence-pair dataset.

The Figure 5.2(left) shows the loss curve of training for the mBART model with the dataset having a count 1000 (i.e. 1K sentence pairs) and the 5.2(right) shows the the loss curve of training for the mBART model with the dataset having a count 4000 (i.e. 4K sentence pairs). Both of the two curves show a regular fall in the loss throughout the course of training along the total epoch up until 20, this conveys the proper learning behaviour of the model and its ability to optimize. The fine-tuning behaviour is reflected in the significant fall in loss within 0 to 15 epochs, as they both approach convergence. For the 1K dataset, loss becomes steady from epoch 17 i.e. starts flattening, while for 4K dataset, loss becomes steady from epoch 9 , both reaching a loss value of around 0.4. Therefore it shows a stronger better regularization capacity with the aid of a larger dataset.

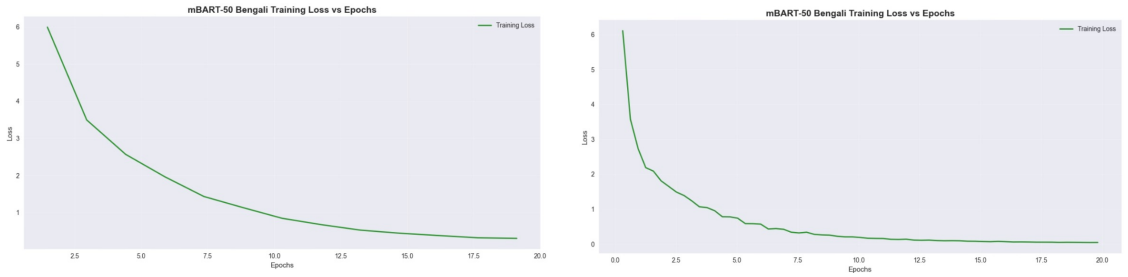


Figure 5.2: Loss convergence for mBART: left = 1000 sentence-gloss pair dataset, right = 4000 sentence-pair dataset.

5.2 Model Fine tuning Results

The results we found show the effectiveness of our data augmentation strategies for Bangla text to gloss translation. Among these models, mBART-50 achieved the best performance on both datasets, with the augmented dataset (4K samples) showing consistent improvements across all BLEU metrics compared to the professional-only dataset of 1K samples. The model achieved a BLEU-4 score of 27.31 and COMET score of 0.8261 on the augmented dataset. mT5 also benefited from data augmentation, with BLEU-4 improving from 17.56 to 20.28 and achieving a COMET score of 0.8602 on the augmented dataset. Unfortunately, NLLB could not be evaluated on the augmented dataset due to computational constraints, though it showed promising results on the manual annotated dataset of 1000 Bangla sentences with a COMET score of 0.907. We also finetuned GPT-4.1-nano to compare its performance to the other models, and it delivered strong results on the augmented dataset, achieving a BLEU-4 score of 22.42 and the COMET score of 0.8913 The consistent performance gains across models validate our multi-faceted augmentation approach combining RAG-based retrieval, rule-based tense conversion, and masking techniques.

Table 5.1: Model performance using BLEU and COMET metrics

Model	**Dataset**	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**	**COMET**
NLLB	Base Dataset (1K)	67.99	40.82	26.49	16.67	0.907
GPT-4.1-nano	Augmented Dataset (4K)	57.11	41.50	31.11	22.42	0.8913
mT5-small	Base Dataset (1K)	51.79	36.26	25.73	17.56	0.8564
mT5-small	Augmented Dataset (4K)	55.33	40.11	29.06	20.28	0.8602
mBART-50	Base Dataset (1K)	56.65	40.96	29.36	21.06	0.7829
mBART-50	Augmented Dataset (4K)	61.09	46.01	35.09	27.31	0.8261

5.3 Comparative Results with RWTH-PHOENIX-2014T

From Table 5.2, the results show our dataset achieved high BLEU-4 and comet score of 27.31 and 0.8261 compared to P-2014T's 21.49 and 0.6673 for mBart. Similarly, for mT5, our dataset has also shown superior Bleu4 and comet results of 20.28 and 0.8602 compared to P-2014T's 18.73 and 0.6205. Overall, our dataset has outperformed P-14T across all metrics for both models. We speculate this is because the sentences in our dataset are of higher quality, less noisy, structured, and compiled

from a specific set of themes rather than P-2014T's random, noisy data compiled from years of weather forecast recordings.

Table 5.2: Evaluation results on P-2014T and our datasets

Dataset	Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	COMET
P-2014T	mBART-50	48.49	35.01	26.85	21.49	0.6673
	mT5-small	54.79	36.47	25.42	18.73	0.6205
Augmented Dataset (4K)	mBART-50	61.09	46.01	35.09	27.31	0.8261
	mT5-small	55.33	40.11	29.06	20.28	0.8602

5.4 Cohen's Kappa Results

The table shows the key evaluation metrics for the effectiveness of our RAG-based data augmentation method. It shows that the validation rate, which measures the percentage of glosses marked as correct by professional signers, was 74.7% for Signer 1 and 76.0% for Signer 2 and the average validation rate is 75.3%. The average quality score, based on a 1–5 scale, was 2.96 from Signer 1 and 3.41 from Signer 2 and an average of 3.19 out of 5. The quality distribution section breaks down how many glosses were rated as high quality, acceptable, or low quality (score 2). Overall, 42.7% of glosses were rated as high quality and 24.3% were acceptable, indicating that most glosses met practical quality standards. Lastly, the reliability is measured using Cohen's kappa, which shows substantial agreement ($\kappa = 0.7489$) for binary validation and fair agreement ($\kappa = 0.3496$) for quality ratings. This supports our conclusion that the evaluations were consistent and trustworthy. As such, we can confidently use the augmented data as a reliable resource for Bangla text-to-gloss translation tasks.

Table 5.3: Key Evaluation Metrics for RAG based data augmentation

Metric	Signer 1	Signer 2	Combined
Validation Rate (%)	74.7	76.0	75.3
Average Quality Score	2.96	3.41	3.19/5.0
Quality Distribution			
High Quality (Score ≥ 4)	35.3%	50.0%	42.7%
Acceptable (Score $= 3$)	26.0%	22.7%	24.3%
Low Quality (Score ≤ 2)	38.7%	27.3%	33.0%
Inter-rater Reliability (Cohen's Kappa)			
Binary Agreement	$\kappa = 0.7489$ (Substantial)		
Quality Agreement	$\kappa = 0.3496$ (Fair)		

Chapter 6

Conclusion

6.1 Research Limitations

While our work lays down the foundation for future work on Continuous Bengali Sign Language Recognition and Translation, it also comes with several limitations that need to be addressed. Firstly, although our dataset of 4000 samples is a valuable extension to such a low-resource setting, such as sign language NLP tasks, it is still relatively limited to ideally train modern transformer architectures to perfectly capture the nuances and morphological patterns that come with converting a Bangla sign language sentence to its corresponding gloss representation. Secondly, manual annotation of Bangla sentence-gloss pairs by an expert is an expensive procedure due to the limited availability of Bangla sign language experts. Due to this, we were only able to cooperate with one expert in our work, so our dataset may include signer bias. Thirdly, sign language is a complex form of communication that is not restricted to hand movements; rather, it also involves facial expressions and body movements to convey different words, emotions, and tones. Glosses, however, as an intermediary, cannot capture facial expressions, and so our dataset lacks a video component to capture these aspects. Additionally, Bangla sign language is limited in vocabulary, due to which many words that the hearing community uses may not have a sign language/gloss representation. So, signers usually break down that word into a set of glosses to express the word. For example, the word "সপ্তাহ" is broken down into "সাত দিন" by signers. As such, our dataset needs to be expanded upon with more such examples for machine translation to capture these special relationships. Finally, names and landmarks do not typically have a dedicated single sign; they are usually spelled out using alphabet-level signing. Glosses like these don't have a specific representation in our dataset which may cause challenges during the training of more advanced sign language translation systems.

6.2 Future Works

The proposed pipeline for generating 3D sign language representations is discussed below. Utilizing the gloss generation procedure in this paper, the next step, is to first create a gloss video dictionary that consists of isolated words, which serve as the key, and their corresponding RGB video representations as their value. From these videos, through the use of State-of-the-art 3D human reconstruction architectures such as SMPLer-X [25] or SMPLify-X [6], generation of 3d human meshes

is possible on a frame-by-frame basis. These models operate by trying to line up 2D keypoints with the 2D projections of the 3D model, thus ensuring accurate and realistic reconstructions from a single RGB image. Once the 3D obj files are generated, they are mapped to their corresponding entries in the 3D sign dictionary, thus completing the 3D dictionary creation process. So the whole procedure, as it stands now, is first to generate gloss sequences from input sentences, then use the glosses to retrieve the corresponding key value pairs from the 3D sign dictionary, and then finally use the series of obj files stored in the value to produce a continuous 3D animation of the signed sentence. We explored this procedure and in fig5.1 we show our findings, but we faced a lot of limitations along the way, such as a lack of a comprehensive available video dataset for sign language words and unstable and jittery animation in Blender due to unsmooth transitions from one sign to the next when animating an entire sign language sentence. We were unable to solve these issues, which we will leave as future work or for other researchers to address. We hope our comprehensive Bangla sentence–gloss dataset will act as a solid base for other researchers to build upon.

Original Image **3D Representation**

Figure 6.1: 3D Mesh generation from an image

6.3 Conclusion

The leveraging of robust data augmentation methods to generate synthetic data i.e. Bengali to sign language gloss sentence pairs as well as fine-tuning pre-trained transformer architectures like mBART, NLLB and MT5 in order to generate corresponding sign language gloss sequence from diverse complete Bengali sentences, our research has introduced the groundwork for promising future research in this particular field. Despite various limitations of our approach and the relative size of our dataset being of small scale, our produced dataset still holds promising prospects for future works in this field. With increased annotated pairs, it will be able to give better results for the translation schemes, thus yielding improved translation

performance in the field of continuous sign language. The augmented dataset could be expanded further with annotated segmented videos for representing those words which do not have any sign i.e. rather conveyed with facial gestures as well as add videos of the Bangla sentences in our dataset performed by real signers. These factors will contribute immensely to the generation of smoother quality of automated text translation to continuous sign language glosses and finally map to accurate corresponding poses.Therefore the work of this paper just does not address the necessity to solve a social issue regarding deaf people but also sets up the foundation to leverage the modern technology to cope up with the linguistic diversity of the World.

Bibliography

[1] M. Alauddin and A. H. Joarder, "Deafness in bangladesh," in *Hearing Impairment: An Invisible Disability How You Can Live With a Hearing Impairment*, J.-I. Suzuki, T. Kobayashi, and K. Koga, Eds. Tokyo: Springer Japan, 2004, pp. 64–69, ISBN: 978-4-431-68397-1. DOI: 10.1007/978-4-431-68397-1_13. [Online]. Available: https://doi.org/10.1007/978-4-431-68397-1_13.

[2] *Janbo Ebar (Pratham Bhag): A book for language development of deaf children*, 1st Edition. Printed on August 2015: Deaf Child Worldwide, 2015, Supported by the National Lottery through the Big Lottery Fund. This book is not for sale. [Online]. Available: https://www.ndcs.org.uk/deaf-child-worldwide/resources-and-research/language-development-books/.

[3] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural sign language translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7784–7793.

[4] A. Conneau, K. Khandelwal, N. Goyal, *et al.*, "Unsupervised cross-lingual representation learning at scale," *CoRR*, vol. abs/1911.02116, 2019. arXiv: 1911.02116. [Online]. Available: http://arxiv.org/abs/1911.02116.

[5] H. R. V. Joze and O. Koller, *Ms-asl: A large-scale data set and benchmark for understanding american sign language*, 2019. arXiv: 1812.01053 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1812.01053.

[6] G. Pavlakos, V. Choutas, N. Ghorbani, *et al.*, " Expressive Body Capture: 3D Hands, Face, and Body From a Single Image," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2019, pp. 10 967–10 977. DOI: 10.1109/CVPR.2019.01123. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR.2019.01123.

[7] C. Raffel, N. Shazeer, A. Roberts, *et al.*, *Exploring the limits of transfer learning with a unified text-to-text transformer*, Oct. 2019. DOI: 10.48550/arXiv.1910.10683.

[8] B. Trezek and C. Mayer, "Reading and deafness: State of the evidence and implications for research and practice," *Education Sciences*, vol. 9, no. 3, 2019, ISSN: 2227-7102. DOI: 10.3390/educsci9030216. [Online]. Available: https://www.mdpi.com/2227-7102/9/3/216.

[9] P. Lewis, E. Perez, A. Piktus, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[10] Y. Liu, J. Gu, N. Goyal, *et al.*, "Multilingual denoising pre-training for neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 726–742, 2020. DOI: 10.1162/tacl_a_00343. [Online]. Available: https://aclanthology.org/2020.tacl-1.47.

[11] Y. Tang, C. Tran, X. Li, *et al.*, "Multilingual translation with extensible multilingual pretraining and finetuning," *CoRR*, vol. abs/2008.00401, 2020. arXiv: 2008.00401. [Online]. Available: https://arxiv.org/abs/2008.00401.

[12] E. J. Hu, Y. Shen, P. Wallis, *et al.*, "Lora: Low-rank adaptation of large language models," *CoRR*, vol. abs/2106.09685, 2021. arXiv: 2106.09685. [Online]. Available: https://arxiv.org/abs/2106.09685.

[13] W. Huang, W. Pan, Z. Zhao, and Q. Tian, "Towards fast and high-quality sign language production," in *Proceedings of the 29th ACM International Conference on Multimedia*, ser. MM '21, Virtual Event, China: Association for Computing Machinery, 2021, pp. 3172–3181, ISBN: 9781450386517. DOI: 10.1145/3474085.3475463. [Online]. Available: https://doi.org/10.1145/3474085.3475463.

[14] M. A. Rahaman, M. I. Hossain, S. Nasrin, M. Rahman, and S. A. Kabir, "An online reversed bangla sign language learning system," in *2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2021, pp. 1–6. DOI: 10.1109/STI53101.2021.9732554.

[15] L. Xue, N. Constant, A. Roberts, *et al.*, "MT5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, *et al.*, Eds., Online: Association for Computational Linguistics, Jun. 2021, pp. 483–498. DOI: 10.18653/v1/2021.naacl-main.41. [Online]. Available: https://aclanthology.org/2021.naacl-main.41/.

[16] M. R. Costa-Jussà, J. Cross, O. Çelebi, *et al.*, "No language left behind: Scaling human-centered machine translation," *arXiv preprint arXiv:2207.04672*, 2022.

[17] E. J. Hwang, J. H. Kim, S. M. Cho, and J. C. Park, *Non-autoregressive sign language production via knowledge distillation*, 2022. arXiv: 2208.06183 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2208.06183.

[18] M. A. Islam, M. A. Islam, A. Karim, *et al.*, "Automatic 3d animated bangla sign language gestures generation from bangla text and voice," in *2022 4th International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2022, pp. 1–6. DOI: 10.1109/STI56238.2022.10103252.

[19] K. K. Podder, M. Chowdhury, A. Tahir, *et al.*, "Bangla sign language (bdsl) alphabets and numerals classification using a deep learning model," *Sensors*, vol. 22, p. 574, Jan. 2022. DOI: 10.3390/s22020574.

[20] Y. Wang, C. Xu, Q. Sun, *et al.*, "Promda: Prompt-based data augmentation for low-resource nlu tasks," *arXiv preprint arXiv:2202.12499*, 2022. [Online]. Available: https://arxiv.org/abs/2202.12499.

[21] Z. Zhang, A. Zhang, M. Li, and A. Smola, "Automatic chain of thought prompting in large language models," *arXiv preprint arXiv:2210.03493*, 2022.

[22] A. Moryossef, M. Müller, A. Göhring, Z. Jiang, Y. Goldberg, and S. Ebling, *An open-source gloss-based baseline for spoken to signed language translation*, 2023. arXiv: 2305.17714 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.17714.

[23] A. Sams, A. H. Akash, and S. M. M. Rahman, "Signbd-word: Video-based bangla word-level sign language and pose translation," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–7. DOI: 10.1109/ICCCNT56998.2023.10306914.

[24] M. K. Ahmed, T. Salam, and L. Y. Pinky, "Bangla sign language classification with deep learning," in *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, 2024, pp. 393–397. DOI: 10.1109/ICEEICT62016.2024.10534441.

[25] Z. Cai, W. Yin, A. Zeng, *et al.*, *Smpler-x: Scaling up expressive human pose and shape estimation*, 2024. arXiv: 2309.17448 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2309.17448.

[26] H. Rubaiyeat, H. Mahmud, A. Habib, and M. K. Hasan, *Bdslw60: A word-level bangla sign language dataset*, Feb. 2024. DOI: 10.13140/RG.2.2.12193.79202.

[27] A. Varanasi, M. Sinha, and T. Dasgupta, "Linguistically informed transformers for text to American Sign Language translation," in *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, A. K. Ojha, C.-h. Liu, E. Vylomova, *et al.*, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 50–56. DOI: 10.18653/v1/2024.loresmt-1.5. [Online]. Available: https://aclanthology.org/2024.loresmt-1.5.

[28] R. Zuo, F. Wei, Z. Chen, B. Mak, J. Yang, and X. Tong, *A simple baseline for spoken language to sign language translation with 3d avatars*, 2024. arXiv: 2401.04730 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2401.04730.

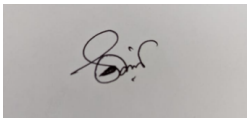
[29] X. Zhang and K. Duh, "Improving sign language gloss translation with low-resource machine translation techniques,"

Appendix A

Expert Testimonial

A.1 Expert Testimonial

This is to testify that Mr. Ariful Islam, an experienced Bangla sign language presenter, professional sign language interpreter and trainer, annotated the 1000 Bangla sentence and gloss sequence pairs in our dataset ensuring linguistic accuracy and cultural appropriateness of the sign language representations. His professional experience helped us in the compilation of the grammatical rules that we used in various augmentation techniques in our augmented dataset creation pipeline. His involvement ensured that our dataset meets professional standards and accurately represents Bangla sign language conventions, lending credibility to our research outcomes.

Signature: 

Mr. Ariful Islam

Date: 23/06/2025