TituLLMs: A Family of Bangla LLMs with Comprehensive Benchmarking

Shahriar Kabir Nahin[1], **Rabindra Nath Nandi**[1], **Sagor Sarker**[1],
Quazi Sarwar Muhtaseem[1], **Md Kowsher**[2], **Apu Chandraw Shill**[1], **Md Ibrahim**[1],
Mehadi Hasan Menon[1], **Tareq Al Muntasir**[1], **Firoj Alam**[3],
[1]Hishab Singapore Pte. Ltd, Singapore, [2]University of Central Florida, USA
[3]Qatar Computing Research Institute, Qatar
{shahriar.nahin, rabindra.nandi, sagor.sarker}@hishab.co, fialam@hbku.edu.qa

Abstract

In this paper, we present *TituLLMs*, the *first* large pretrained Bangla LLMs, available in 1b and 3b parameter sizes. Due to computational constraints during both training and inference, we focused on smaller models. To train *TituLLMs*, we collected a pretraining dataset of approximately $\sim$ 37 billion tokens. We extended the Llama-3.2 tokenizer to incorporate language- and culture-specific knowledge, which also enables faster training and inference. There was a lack of benchmarking datasets to benchmark LLMs for Bangla. To address this gap, we developed *five benchmarking datasets*. We benchmarked various LLMs, including *TituLLMs*, and demonstrated that *TituLLMs* outperforms its initial multilingual versions. However, this is not always the case, highlighting the complexities of language adaptation. Our work lays the groundwork for adapting existing multilingual open models to other low-resource languages. To facilitate broader adoption and further research, we have made the *TituLLMs* models and benchmarking datasets publicly available.[1]

1 Introduction

The rapid advancements in large language models (LLMs) have reshaped the field of AI, showcasing remarkable versatility across numerous tasks (Brown et al., 2020; Ouyang et al., 2022; Achiam et al., 2023; Chowdhery et al., 2023a). These models demonstrate not only an ability to perform various NLP tasks but also an intriguing potential for self-assessment and continuous improvement (Liu et al., 2023b; Fu et al., 2023; Chiang et al., 2023).

Despite these advancements, LLM development—both open and closed—has predominantly focused on multilingual models, with a stronger emphasis on high-resource languages (Achiam et al., 2023; Touvron et al., 2023). While some models have extended coverage to medium- and low-resource languages (Le Scao et al., 2023; Üstün et al., 2024; Yang et al., 2024; Team, 2024), their representation remains limited. Although some initiatives have aimed to train language-centric LLMs (Sengupta et al., 2023; Team et al., 2025), these efforts remain scarce due to the high costs associated with computational resources and data collection. Consequently, recent research has shifted towards adapting existing LLMs for new languages (Levine et al., 2024; Team et al., 2025).

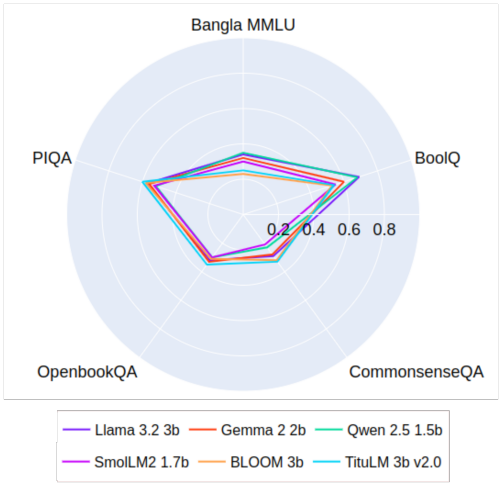

Figure 1: Performance scores per category TituLLM-3B and five other models in 5-shot setting.

Similarly, in the context of benchmarking LLMs, most efforts have primarily focused on high-resource languages (Bang et al., 2023; Ahuja et al., 2023), while low-resource languages, such as Bangla, have received limited attention (Kabir et al., 2024; Zehady et al., 2024; Bhattacharjee et al., 2023). Zehady et al. (2024) developed LLMs for Bangla using Llama, leveraging only the Bangla subset of CulturaX (Nguyen et al., 2024), which consists of 12.4 million diverse Bangla news articles. They further fine-tuned their model on 172k instruction samples from subsets of the Alpaca (Taori et al., 2023) and OpenOrca (Lian et al., 2023) datasets, which were translated from English.

[1]https://github.com/hishab-nlp/titulm

Findings of the Association for Computational Linguistics: ACL 2025, pages 24922–24940

Their models were benchmarked on 120 queries across nine different generation tasks.

Addressing this gap is crucial, as linguistic and cultural diversity significantly impact language understanding and generation. Therefore, in this study, we focus on adapting existing LLMs (e.g., Llama) by expanding the model's vocabulary and performing continual pretraining. This process required extensive data collection from diverse sources. Given the relatively low availability of digital content in Bangla, we also developed synthesized datasets to supplement our training data.

Benchmarking LLM capabilities for Bangla remains challenging due to the lack of specialized datasets, particularly in areas such as world knowledge and commonsense reasoning. Although some efforts have been made to generate such datasets through translation (Lai et al., 2023), they remain limited in scope. To address this gap, we have developed several native and translated datasets. ***Compared to*** Zehady et al. (2024), our pretraining corpus is significantly larger ($\sim 37b$ tokens) and is benchmarked on five different datasets covering world knowledge and commonsense reasoning, with a total dataset size of 132k entries.

- We developed and released two models, *TituLLMs*, adapted from Llama 3.2, which will enable future research.
- We provide a complete data collection recipe for pretraining LLMs including sources, approaches to synthetic data generation.
- We extended tokenizer to ingest language specific knowledge.
- We developed ***five datasets*** to benchmark LLMs capabilities in terms of world knowledge, commonsense reasoning, and reading comprehension. Such datasets will serve as a first step to Benchmark LLMs for Bangla.
- We proposed a novel translation techniques (*expressive semantic translation*) that helps to develop high quality benchmarking dataset.
- Using the benchmaked datasets we benchmark various LLMs including *TituLLMs* comparing performance across models to assess understanding of Bangla language.

Our study reveals several interesting findings:

Vocabulary Extension: We explore the impact of vocabulary extensions on the base Llama Tokenizer by increasing the number of new tokens from 32K to 96K in increments of 16K. We found that average tokens per word (TPW) decreases as the number of tokens increases up to a certain point, after which it declines only minimally. Therefore, when adding new tokens, we must also consider the fertility rate to balance the trade-off between training and inference.

Commonsense Capability: *TituLLMs* demonstrates strong commonsense knowledge but has limited capability in world knowledge (e.g., Bangla MMLU). Further training with instruction fine-tuning may enhance its performance in this area.

2 Pretraining Data

Pretraining data for Bangla is very limited compared to very high quality data available for English and other high resource languages (Penedo et al., 2023; Soldaini et al., 2024). Hence, we needed collect pretraining dataset for training *TituLLMs*. We have compiled a substantial Bangla raw dataset from a diverse range of sources, encompassing both formal and informal linguistic styles. The dataset is primarily derived from three key sources: web documents, books, and synthetically generated text. The synthetically generated data includes translated data, transliterated data, audio transcripted data, etc. An outline of our data collection and preprocessing pipeline is shown in Figure 7. The final high-quality dataset amounts to approximately 268 GB, with 22 GB designated as a validation set, sampled proportionally to maintain the heterogeneity of the data. In total, the corpus contains $\sim$ 37 billion tokens, optimized for training and evaluation across various Bangla language processing applications. In Table 4 (in Appendix) and in Figure 3, we report the distribution of tokens for different sources.

2.1 Web Documents

We curated the *Common Crawl* (Raffel et al., 2020) dataset and followed multiple steps to extract and clean the final dataset, as illustrated in Figure 7. Below, we briefly discuss each step.

SQL Query Engine: Using Amazon Athena,[2] we queried the vast Common Crawl dataset to isolate Bangla-specific HTML data and URLs. We applied filtering based on content language, URL patterns indicative of Bangla domains (e.g., `.gov.bd`), and host information, covering data from 2017 to 2024.

Text Extraction: We used *Trafilatura* (Barbaresi, 2021) tool for its effectiveness in extracting structured, clean text from HTML. This step preserved

[2] https://aws.amazon.com/athena/

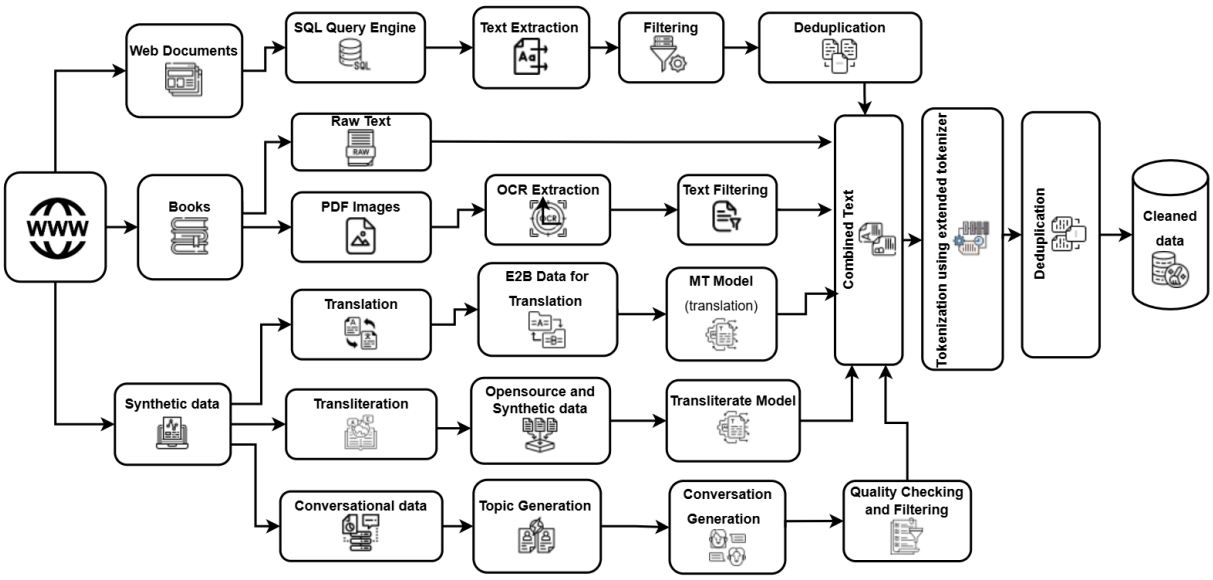

Figure 2: Overview of the pretraining data collection and preprocessing pipeline – A workflow illustrating the steps involved in gathering, filtering, and preparing data for LLM pretraining.

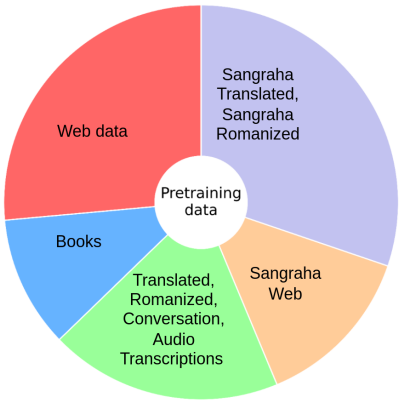

Figure 3: The pretraining dataset consists of $\sim 37b$ billion tokens distributed across various sources: Web (9.8b), Books (4b), Synthetic Data (7.06b), Sangraha Web (5b), and Sangraha Synthetic (10.94b). Synthetic Data includes Translated (1.47b), Romanized (3.87b), Conversation (0.42b), and Audio Transcription (1.30b), while Sangraha Synthetic comprises Translated (4.26b) and Romanized (6.68b).

the structure of text patterns while minimizing extraneous noise, ensuring the retention of original content nuances.

Filtering: To assess the usability and relevance of the extracted text, we generated 18 distinct handcrafted rules specifically designed for the Bangla language. These rules evaluated various aspects of text quality, including sentence completeness, document structure, and content appropriateness, leveraging Bangla-specific tokenizers for a proper filtering process. The details of these rules are discussed in Section A.2. After extraction, documents were assessed against predefined thresholds associated with each rule. Based on the threshold we filtered documents that did not pass the criteria.

Deduplication: We applied the MinHash deduplication algorithm (Kocetkov et al., 2023) to the filtered web dataset to eliminate redundant content.

2.2 Books

We have compiled a diverse collection of open-source Bangla books, primarily in PDF format, spanning a broad temporal range from historical to contemporary works. Below we briefly discuss the steps taken to extract the text from the book collection.

Raw Text: For digitally available texts, we directly extract the machine-readable content, requiring minimal processing due to the already high quality of these formats.

PDF Images: To digitize a vast collection of non-digital and older texts from books, we utilize two leading Optical Character Recognition (OCR) systems: Google OCR[3] and Tesseract[4]. We used Tesseract to reduce the cost. These texts, often derived from sources that have deteriorated over time or originated in non-digital formats, pose significant challenges in terms of quality and legibility. To extract high-quality text, we implement carefully designed techniques comprising several steps.

Text extraction using Google OCR: For the majority of books, we utilized Google OCR to extract

[3] https://cloud.google.com/use-cases/ocr
[4] https://github.com/tesseract-ocr/tesseract

text. Although the OCR accuracy was generally high for more recent books, the quality varied significantly for older books. Identifying and filtering out poor-quality text from a large and diverse collection proved to be a challenging task. To address this issue, we applied several quality-control techniques: *(i)* using KenLM (Heafield, 2011) to filter noisy text based on ranking, *(ii)* evaluating the average number of words and sentences per page, and *(iii)* calculating the percentage of correct Bangla words. Details are discussed in Section A.3.

Document Segmentation and Tesseract OCR: We performed document segmentation alongside OCR for a smaller portion of the collected books. Initially, we trained a YOLO segmentation model on a Bangla document segmentation dataset (Shihab et al., 2023) to identify and classify different components of the documents (e.g., text boxes, tables, paragraphs, and images). We removed complex sections such as tables and images, as they were not relevant to the text extraction process. Subsequently, we applied Tesseract OCR to the remaining document text and repeated the filtering process outlined earlier. In addition to the filtering steps, we introduced an additional measure: we calculated the number of words with more than 80% confidence and set a threshold at the 95th percentile to filter out low-quality text. After applying all these processes, 50% of the initial data was retained.

2.3 Synthetic Data

Due to the low representation of digital content in Bangla, we have developed a large-scale synthetic dataset for Bangla, which include transcription, translation and transliterated data.

Transcribed Text: We collected conversational and spoken language data transcribed using the Bangla Automatic Speech Recognition (ASR) system (Nandi et al., 2023). This system enables us to capture various colloquial and regional linguistic variations in Bangla. We collected approximately 56k hours of speech data from diverse online sources. All collected speech data were transcribed using the ASR system.

Translation Data: To collect English-to-Bangla translated data, we trained an NLLB-based (600M-Distilled) model (Team et al., 2022) with the goal of developing a smaller, language-pair-specific (en-bn) model. We decided to train a translation model because our observations indicate that currently available multilingual models, such as Llama-3.1-8B-Instruct, have limited capability for Bangla-specific generation tasks. However, they have shown superior performance in English-specific generation.

For training the en-bn machine translation (MT) model, we collected open-source translation data from various platforms, including BanglaNMT (Hasan et al., 2020a) and Samanantar (Ramesh et al., 2022). Furthermore, we generated synthetic bn-en translation pairs using Bangla news sources and Wikipedia as source data, employing Llama-3.1-8B-Instruct (Touvron et al., 2023) for target data generation. We selected Llama for this task due to its superior English-language capabilities. Using this approach, we created a dataset comprising approximately 60 million translation pairs, which we then used to train the NLLB model. We have named this model as *Titu-Translator*. On our in-house test dataset, the BLEU score is 37.6.

Evaluation using Scalar Quality Metrics (SQM). We have also evaluated our model using SQM (Lankford et al., 2022). It is a structured and interpretable human evaluation of machine translation. Table 1 presents the results of human evaluation conducted by two evaluators across five distinct domains: Business, Entertainment, Politics, Tech, and Sport. Among the evaluated models, *Titu-Translator* demonstrates strong performance, achieving an overall average score of 4.99, which places it among the top-performing models. It performs particularly well in the Tech and Entertainment domains, with scores up to 5.30 from Human-2 and 4.95 from Human-1. While Indic-Trans-2 slightly outperformed *Titu-Translator*, we chose *Titu-Translator* for synthetic pretraining due to its lightweight design and alignment with Bangladeshi linguistic styles. Appendix C describes more about the evaluation process.

Once the model has been trained, we have used it to translate a corpus of English news articles[5] into Bangla.

Additionally, we added translated data from the *Sangraha* dataset (Khan et al., 2024) to our corpus. The synthetic data coming from the *Sangraha* dataset is generated with Indic-Trans-2, which is reported to be the top-performing model for English to Bangla translation in their report (Gala et al., 2023). It is also reflected in our SQM evaluation.

[5] https://www.kaggle.com/datasets/davidmckinley/all-the-news-dataset

Model	Business		Entertainment		Politics		Tech		Sport		Average
	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	
Titu Translator	4.8	5.3	4.85	**5.4**	4.8	5.05	4.95	**5.3**	4.75	4.65	4.99
nllb-600M	4.45	5.1	4.65	5.0	4.75	4.85	4.85	5.05	4.55	4.2	4.75
csebuetnlp-t5	4.55	4.95	4.65	4.8	**4.85**	4.9	4.6	4.9	4.65	4.65	4.75
Indic-Trans-2	**4.95**	**5.35**	4.85	**5.35**	**4.85**	**5.35**	**5.0**	4.9	**5.0**	**5.05**	**5.07**
Google Translator	4.6	4.75	**4.89**	4.65	4.65	4.85	4.95	4.8	**5.0**	4.55	4.77

Table 1: Human (H) evaluation by two annotators across five domains for each translation model.

Transliteration Data: The use of romanized text is very common in everyday communication for Bangla (Fahim et al., 2024). To address this, we have developed a Bangla-to-Romanized dataset by training an NLLB-based (600M-Distilled) model. For model training, we collected transliteration pairs from the *Sangraha* dataset and generated additional synthetic transliteration pairs using the GPT-4 model (Achiam et al., 2023). We then used this dataset to train the NLLB-based transliteration model. The BLEU score for this model is 65.1, as evaluated on an in-house test dataset. We then used this model to create the transliteration dataset by selecting a small subset of collected Bangla Wikipedia articles.

Conversational Data: To enhance the model with conversational capabilities, we enriched our dataset by incorporating conversational data. We have crawled topics (e.g., *"Rabindranath Tagore's contributions to Bengali art"*) from Wikipedia and Banglapedia on which we generated conversations between two agents. To achieve this we have developed an agentic system where two agents interact with each other on a given topic. In Appendix Table 7, we have provided examples of topic, roles, and a detail of the prompt. An example conversation is provided in Appendix Figure 7. The average number of turns per conversation is 8. In total, we added $\sim$1 million conversations to the dataset.

2.4 Sangraha Dataset

Additionally, we enriched our dataset by integrating the open-source Sangraha dataset. It is the largest high-quality, cleaned Indic language corpus. We incorporated the Bangla web data portion of the dataset into our training set.

3 Pretraining

3.1 Tokenizer Training

We developed a custom tokenizer for Bangla text using Tiktoken[6], which employs Byte Pair Encoding (BPE) (Sennrich et al., 2016) for tokenization. To train this tokenizer, we sampled 48 GB of data from our pretraining Bangla corpus. Additionally, we modified the original Tiktoken codebase to enhance its efficiency and better accommodate the morphological complexities of Bangla. After training multiple tokenizers on the same subset, we merged each newly trained tokenizer with the existing Llama-3.2 tokenizer. This merging process aimed to preserve the strengths of the original tokenizer while integrating domain-specific vocabulary and improving segmentation for Bangla. To evaluate the performance of each merged tokenizer, we computed the average tokens per word (TPW) on a separate 1 GB sample from the original corpus. Table 5, in Appendix, summarizes the TPW values for both the original Llama-3.2 tokenizer and the newly trained tokenizers. A lower TPW generally indicates more efficient segmentation, which reduces sequence lengths and may enhance downstream model performance.

We trained five tokenizers with different vocabulary sizes, as presented in Table 5. Each of these tokenizers was then merged with the Llama-3.2 tokenizer to create five new tokenizers. Notably, the Llama-3.2 tokenizer exhibits a very high TPW value, which affects its performance for Bangla. In contrast, the newly developed tokenizers demonstrate significantly lower TPW values.

The table also shows that increasing the vocabulary size of the new tokenizers generally results in a lower TPW count. However, the relationship between vocabulary size and TPW is not strictly linear. While TPW decreases with larger vocabularies, the reduction becomes less significant for tokenizers with very large vocabulary sizes.

3.2 Model Architecture

We have modified Llama-3.2-1b and Llama-3.2-3b models according to the merged tokenizers. As too many new tokens will increase the model's size and the training complexity, we modified the models according to Llama-3.2-plus-48K tokenizer. We

[6] https://github.com/openai/tiktoken

added extra embedding vectors in the embedding layer and modified the lm-head according to the vocabulary size.

3.3 Pretraining

After modifying the models, we pre-trained them on our full dataset using LlamaFactory (Zheng et al., 2024). Both models were trained with a context length of 4096, with packing enabled for maximum efficiency. Training for one epoch required 1,750 H100 GPU hours.

4 Evaluation

4.1 Evaluation Setup

For evaluation, we utilized the lm-evaluation-harness.[7] We used normalized accuracy as a metric. Our assessment focuses on key aspects such as *knowledge* and *reasoning*.

4.2 Banchmarking Datasets

We benchmarked *TituLLMs* alongside other popular LLMs using five newly prepared evaluation datasets. The dataset is composed of multiple subsets of the benchmarking set, including Bangla MMLU (87,869 entries), Piqa BN (17,177 entries), CommonsenseQA BN (10,962 entries), OpenBookQA BN (5,944 entries), and BoolQ BN (1,976 entries). This distribution highlights the significant dominance of the Bangla MMLU subset within the overall evaluation dataset.

Below, we describe the development process for each dataset. Table 2 presents the distribution and splits of each dataset.

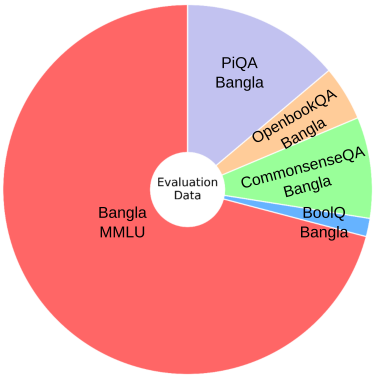

Figure 4: Distribution of an benchmarking dataset totaling $\sim$ 124 entries.

Bangla MMLU: We curated multiple-choice questions from various open-source educational websites and textbooks, inspired by the original MMLU dataset (Hendrycks et al., 2021). The dataset includes multiple-choice questions from different Bangladeshi exams, such as job exams, the Bangladesh Civil Service Exam, and undergraduate admission exams.

Figure 5 provides a detailed breakdown of the Bangla MMLU dataset, which contains 87,869 questions spanning multiple educational categories. These include University Admission (47,394), Higher Secondary (25,437), Job Exams (9,122), Medical Admission (3,764), and Engineering Admission (2,152). The dataset reflects a diverse range of question types relevant to various levels of academic and professional assessments, making it a comprehensive benchmark for evaluating LLMs in Bangla educational contexts.

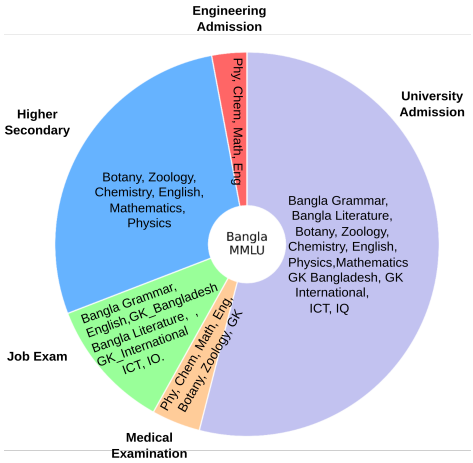

Figure 5: The Bangla MMLU dataset comprises a total of 87,869 questions distributed across various educational categories: University Admission (47,394), Higher Secondary (25,437), Job Exams (9,122), Medical Admission (3,764), and Engineering Admission (2,152).

CommonsenseQA Bangla (CSQA): We translated the CommonsenseQA dataset (Talmor et al., 2019) into Bangla using our custom translation-based approach, *Expressive Semantic Translation (EST)*. This method generates multiple translations for a sentence and iteratively refines them to select the most suitable version. More details on this approach are discussed in Appendix Section D.

OpenBookQA Bangla (OBQA): This dataset is a Bangla translation of the OpenBookQA dataset (Mihaylov et al., 2018), which is designed to test a model's ability to apply elementary science knowledge to answer open-domain multiple-choice questions. We translated OpenBookQA into Bangla using our EST method.

PIQA Bangla (PIQA): This dataset is a Bangla

[7] https://github.com/EleutherAI/lm-evaluation-harness

translation of the Physical Interaction: Question Answering (PIQA) dataset (Bisk et al., 2020), which evaluates a model's understanding of everyday physical reasoning and common-sense interactions. PIQA consists of multiple-choice questions requiring knowledge about how objects interact in the real world, such as choosing the most practical way to perform a given task. For the translation, we used our EST method.

BoolQ Bangla (BoolQ): This dataset is inspired by BoolQ (Clark et al., 2019), a reading comprehension benchmark that evaluates a model's ability to answer yes/no questions based on a given passage. The dataset consists of triplets in the form of (question, passage, answer). Passages were sourced from Bangla Wikipedia, Banglapedia, and news articles, ensuring a diverse range of topics and contexts. To generate high-quality questions and answers, we leveraged GPT-4.

Dataset	Method	Train	Val.	Test	Dev
Bangla MMLU	Manual	-	72,944	**14,750**	175
BoolQ	GPT-4	815	432	**729**	-
CommonsenseQA	EST	9,741	**1,221**	-	-
OpenBookQA	EST	4,947	500	**497**	-
PIQA	EST	15,339	**1,838**	-	-

Table 2: Data splits and distribution of the Benchmark dataset. Val.: Validation.

5 Results and Discussion

We evaluate each model in 0-shot and 5-shot settings to assess their few-shot adaptability. Table 3 presents the detailed results for all models,[8] including the *TituLLMs* variants. Table 3 shows the accuracy of various models with less than or equal to 3b parameters and the GPT-davinci-002 model.

Bangla MMLU: In the 0-shot setting, both the TituLLM-1b and TituLLM-3b models score 0.25, placing them in the mid-range relative to other 1b–3b models in the Bangla MMLU benchmark. Neither model shows gains when moving to the 5-shot setting (both remain at 0.25), suggesting that additional examples do not substantially improve performance for this specialized knowledge benchmark. It is possible that the domain-specific knowledge required for MMLU-like tasks is not adequately captured by our model. The primary reason behind this can be the lack of extensive pertaining. We have trained our model with only $\sim 37b$ tokens for one epoch. As a result, the model could not capture the full knowledge base (Hoffmann et al., 2022). Another reason behind this can be diversity in datasets. For example, Llama and Qwen models are trained on a high volume of English datasets that have helped these models to have a better knowledge base.

Model	S	BN MMLU	BoolQ	CSQA	OBQA	PIQA
davinci	0	0.30	0.53	0.22	0.30	0.52
	5	-	-	-	-	-
Llama-3.2-1b	0	0.28	0.53	0.23	0.32	0.53
	5	0.28	0.58	0.23	0.32	0.54
Llama-3.2-3b	0	**0.33**	0.53	0.26	0.32	0.57
	5	0.34	**0.69**	0.29	0.32	0.57
Gemma-2-2b	0	0.29	0.56	0.26	**0.34**	0.56
	5	0.32	0.60	0.28	0.33	0.56
Qwen-2.5-0.5b	0	0.30	0.53	0.21	0.31	0.54
	5	0.31	0.58	0.22	0.30	0.53
Qwen-2.5-1.5b	0	**0.33**	**0.62**	0.23	0.29	0.53
	5	**0.35**	0.68	0.23	0.30	0.52
SmolLM2-135m	0	0.23	0.53	0.22	0.31	0.52
	5	0.23	0.51	0.21	0.30	0.52
SmolLM2-360m	0	0.25	0.53	0.20	0.30	0.54
	5	0.24	0.52	0.21	0.29	0.53
SmolLM2-1.7b	0	0.29	0.53	0.22	0.31	0.53
	5	0.30	0.55	0.21	0.30	0.53
BLOOM-560m	0	0.23	0.53	0.26	0.31	0.54
	5	0.23	0.53	0.26	0.28	0.54
BLOOM-1b1	0	0.26	0.56	0.27	0.31	0.54
	5	0.23	0.58	0.27	0.31	0.55
BLOOM-1b7	0	0.27	0.53	0.27	0.32	0.55
	5	0.27	0.59	0.30	0.31	0.56
BLOOM-3b	0	0.26	0.53	0.27	0.33	**0.58**
	5	0.23	0.53	0.32	0.31	0.58
BongLLaMA-3.2-1b	0	0.25	0.53	0.22	0.33	0.52
	5	0.26	0.53	0.24	0.31	0.53
BongLLaMA-3.2-3b	0	0.30	0.53	0.21	0.27	0.51
	5	0.33	0.54	0.20	0.29	0.50
TituLLM-1b-v2.0	0	0.25	0.53	0.26	0.32	**0.58**
	5	0.25	0.51	0.28	0.33	0.57
TituLLM-3b-v2.0	0	0.25	0.53	**0.28**	0.32	**0.58**
	5	0.25	0.54	**0.33**	**0.35**	**0.60**

Table 3: Benchmark results (normalized accuracy) across models and datasets for 0-shot and 5-shot settings. S: Shots, BN MMLU: Bangla MMLU, davinci: GPT-davinci-002,

BoolQ: The BoolQ dataset measures the performance of the model for yes/no question-answering in Bangla. TituLLM-1b achieves 0.53 in the 0-shot setting but drops slightly to 0.51 in the 5-shot setting. In contrast, TituLLM-3B moves from 0.53 (0-shot) to 0.54 (5-shot). However, Llama-3b and Qwen-2.5-1.5b have done much better in this task. As the context length for all BoolQ data matched that of a News document, TituLLM's performance may drop on long contexts. This suggests further pretraining should target longer contexts (Chowdhery et al., 2023b; Kaplan et al., 2020).

[8]GPT (OpenAI, 2023), Llama (Touvron et al., 2023), Gemma (Team et al., 2024), Qwen (Chu et al., 2024), SmolLM2 (Allal et al., 2025), BLOOM (Le Scao et al., 2023), BongLLaMA (Zehady et al., 2024).

Figure 6: Example of tokenization of Llama and TituLLM tokenizers.

CSQA, OBQA, and PIQA: Commonsense reasoning tasks often challenge smaller-scale language models. The accuracy of the 3B variant of TituLLM, starts at 0.28 (0-shot) and exhibits a more pronounced jump to 0.33 (5-shot) which is the maximum among all models. TituLLM-1b has also shown decent performance on the CSQA dataset.

OBQA requires both textbook knowledge and reasoning. Similar to CSQA, TituLLM-3b shows superior performance in this dataset 0.35. Both the dataset's results suggest that TituLLM's reasoning capability is better than other base models.

PIQA tests physical commonsense knowledge. TituLLM-3b model shows better performance in this task too with an accuracy of 0.60. By observing the results on the CSQA, OBQA, and PIQA datasets we can say that the model has captured Bangla Language specific reasoning well in spite of being trained with a smaller dataset than others but the results from MMLU and BoolQ shows the impact of limited training.

Performance of Tokenizer: The superior performance of our models in reasoning tasks is mainly an impact of our extended tokenizer. To justify this, we can observe the results of the BongLLaMa models. These models are continual pretrained models with existing open-source Bangla text corpus. If only the dataset could improve the performance then that would be reflected in BongLLaMA models. However, we observe that they are performing similarly to Llama models. To have an interpretation of our extended tokenizer's performance we can look into Figure 6. The figure shows Llama tokens and TituLLM tokens for a simple sentence in Bangla with two of the most common words. We observe that Llama tokenizer splits the text into character and byte levels. On the other hand, TituLLM tokenizes the sentence into word or subword levels. As a result, TituLLM can deliver more meaningful tokens than Llama for Bangla text. This is an important advantage of TituLLM that not only enables TituLLM to perform better with smaller datasets but also ensures low latency during inference.

6 Related Work

Pretraining: Pretraining LLMs on Bangla has involved the development of specialized models like BongLLaMA (Zehady et al., 2024), which has been adapted from Llama to better understand and generate Bangla text. The pretraining phase typically leverages large-scale Bangla corpora to improve the model's foundational understanding of the language's syntax and semantics. For instance, Zehady et al. (2024) focused on developing a robust model by pretraining on diverse Bangla data sources, significantly improving the model's performance on native text. Similar efforts have been made in previous research, such as BanglaBERT (Bhattacharjee et al., 2022) and SahajBERT (Diskin et al., 2021), where models underwent extensive pretraining on curated Bangla datasets to better capture linguistic nuances.

Enhancing Tokenization: The evolution of token adaptation in NLP has progressed from linguistic cues and statistical methods (Creutz and Lagus, 2006; Luong et al., 2013; Zhang et al., 2023) to phrase-level segmentation (Koehn et al., 2007, 2003). The rise of deep learning shifted the focus to subword-level segmentation, enhancing the handling of rare words (Sennrich et al., 2016; Kudo, 2018; Kudo and Richardson, 2018). More recent efforts emphasize integrating specialized vocabularies into pre-trained LLMs, prioritizing tokenization quality and cost-effectiveness (Ahia et al., 2023; Zhang et al., 2023, 2024; Tejaswi et al., 2024). Liu et al. (2023a) propose a model-agnostic approach for adapting extended vocabularies to LLMs by integrating task-specific vocabularies, prioritizing new tokens, and initializing their embeddings using averaged subword representations. Cui et al. (2023) extend Llama's existing vocabulary by incorporating an additional 20k Chinese tokens, enhancing its ability to understand and generate Chinese text. Chiappe and Lennon (2024) develop an adaptive tokenization algorithm that implements a dynamic tokenization dictionary within the Llama model, updating tokens in real-time based on frequency and contextual relevance.

Cross-Lingual Model Adaptation: Cross-lingual transfer enables models trained in one language to adapt to others without retraining from scratch. Key adaptation techniques include embedding initialization, transliteration, and vocabulary extension. Jaavid et al. (2024) used

transliteration to convert non-Latin languages into Latin scripts for better knowledge transfer. Lai et al. (2024) trained a model on millions of target-language tokens without vocabulary extension, achieving performance comparable to models trained on billions of tokens. However, tokenization mismatches reduced inference efficiency. Studies by Csaki et al. (2023); Cui et al. (2023); Raffel et al. (2020); Lin et al. (2024) found that vocabulary extension improves performance while reducing computational inefficiencies. Tejaswi et al. (2024) further explored language-specific LLMs, highlighting trade-offs in adaptation for low-resource languages. Their findings emphasize that while vocabulary expansion enhances efficiency, selecting the right base model and vocabulary size is crucial.

Benchmarking and Evaluation. Evaluating LLMs requires benchmarking datasets that assess a wide range of capabilities and tasks. For Bangla, most existing datasets focus on standard NLP tasks. The BanglaNLG benchmark dataset (Bhattacharjee et al., 2023) addresses this by integrating six distinct datasets designed to evaluate various aspects of natural language generation (NLG) in Bangla, including Machine Translation, Text Summarization, Question Answering, Multi-turn Dialogue, News Headline Generation, and Cross-lingual Summarization. Beyond NLG, the Region-Specific Native-QA dataset (Hasan et al., 2025) was developed to assess the question-answering capabilities of leading LLMs, such as GPT-4o, GPT-4, Gemini, Llama-3, and Mistral. By focusing on regionally relevant queries, this dataset ensures that models are tested in real-world Bangla language contexts. For a broader evaluation of LLMs across multiple tasks, BenLLM (Kabir et al., 2024) provides the most comprehensive comparison of model performance. This study benchmarks LLMs against other pretrained models using datasets from diverse sources, offering insights into their strengths and limitations across various NLP tasks. Other relevant benchmarking efforts include sentiment analysis (Hasan et al., 2024), question answering (Hasan et al., 2025; Shafayat et al., 2024; Alam et al., 2025), summarization (Tanjila et al., 2025), and cultural understanding (Kabir et al., 2025).

There remains a notable scarcity of benchmarking datasets for evaluating the emergent capabilities of LLMs, such as world knowledge and cognitive reasoning. To address this gap, we introduce ***five benchmarking datasets***, each designed to assess specific competencies, including world knowledge and commonsense reasoning.

7 Conclusion

In this study, we present the *first* pretrained Bangla LLMs, *TituLLMs*, trained on $\sim$ $37b$ tokens by adapting Llama-3.2 models. We extended the tokenizer to incorporate language- and culture-specific knowledge, which also enable faster training and inference. Pretraining data collection remains challenging for languages with low digital representation. To address this, we provide a comprehensive approach, including raw web data collection, translation, and synthetic data generation. Given the lack of LLM-based benchmarking datasets, we developed *five datasets* comprising $137k$ samples, covering both *knowledge* and *reasoning*. The benchmarking dataset includes manually curated samples as well as a novel translation-based (EST) approach. Using these datasets, we benchmarked various LLMs, including *TituLLMs*, demonstrating its superior performance in reasoning tasks without instruction tuning. Future work includes collecting larger pretraining datasets and fine-tuning with instruction-based datasets.

8 Limitations

There are two limitations in this work. Firstly, despite the improvements observed in the 3b variant, the model's performance on long contexts remains suboptimal, suggesting the need for further enhancement in handling extended sequences. Secondly, while the current models are trained solely on Bangla text, their performance could benefit from incorporating larger, English-centric datasets. This could facilitate better knowledge leveraging and potentially improve low-resource language performance, indicating a direction for future research. Since there is a lack of instruction tuning data in Bangla, we do not explore the full potential of instruction tuning, which could have further improved the models' performance on specialized tasks and domain adaptation.

Ethical Consideration

We do not anticipate any ethical concerns in this study. All datasets used were collected from publicly available sources, ensuring compliance with ethical research standards. No personally identifiable information (PII) was gathered or utilized

in the development of our models. While we do not foresee any potential risks arising from the outcomes of this study, we strongly encourage users of the released models to adhere to responsible AI usage guidelines. This includes avoiding the generation or dissemination of harmful, misleading content and ensuring that the models are employed in ethical and socially beneficial applications.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David R Mortensen, Noah A Smith, and Yulia Tsvetkov. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Firoj Alam, Md Arid Hasan, Sahinur Rahman Laskar, Mucahid Kutlu, and Shammur Absar Chowdhury. 2025. NativQA Framework: Enabling llms with native, local, and everyday knowledge. *arXiv preprint arXiv:2504.05995*.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, et al. 2025. SmolLM2: When smol goes big–data-centric training of a small language model. *arXiv preprint arXiv:2502.02737*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675—-718, Indonesia. Association for Computational Linguistics.

Adrien Barbaresi. 2021. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131, Online. Association for Computational Linguistics.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M Sohel Rahman, and Rifat Shahriyar. 2022. Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, and Rifat Shahriyar. 2023. Banglanlg and banglat5: Benchmarks and resources for evaluating low-resource natural language generation in bangla. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 726–735.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90% ChatGPT quality. https://vicuna.lmsys.org. Accessed 14 April 2023.

Harold Chiappe and Gabriel Lennon. 2024. Optimizing knowledge extraction in large language models using dynamic tokenization dictionaries. *OSF Preprints*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023a. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023b. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Mathias Johan Philip Creutz and Krista Hannele Lagus. 2006. Morfessor in the morpho challenge. In *PASCAL Challenge Workshop on Unsupervised segmentation of words into morphemes*.

Zoltan Csaki, Pian Pawakapan, Urmish Thakker, and Qiantong Xu. 2023. Efficiently adapting pretrained language models to new languages. *arXiv preprint arXiv:2311.05741*.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

Michael Diskin, Alexey Bukhtiyarov, Max Ryabinin, Lucile Saulnier, Anton Sinitsin, Dmitry Popov, Dmitry V Pyrkin, Maxim Kashirin, Alexander Borzunov, Albert Villanova del Moral, et al. 2021. Distributed deep learning in open collaborations. *Advances in Neural Information Processing Systems*, 34:7879–7897.

Md Fahim, Fariha Shifat, Fabiha Haider, Deeparghya Barua, Md Sourove, Md Ishmam, and Md Bhuiyan. 2024. BanglaTLit: A benchmark dataset for back-transliteration of romanized bangla. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14656–14672.

Yao Fu, Litu Ou, Mingyu Chen, Yuhao Wan, Hao Peng, and Tushar Khot. 2023. Chain-of-thought hub: A continuous effort to measure large language models' reasoning performance. *arXiv preprint arXiv:2305.17306*.

Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Preprint*, arXiv:2305.16307.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Derek Greene and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 377–384, New York, NY, USA. Association for Computing Machinery.

Md. Arid Hasan, Shudipta Das, Afiyat Anjum, Firoj Alam, Anika Anjum, Avijit Sarker, and Sheak Rashed Haider Noori. 2024. Zero- and few-shot prompting with LLMs: A comparative study with fine-tuned models for Bangla sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17808–17818, Torino, Italia. ELRA and ICCL.

Md Arid Hasan, Maram Hasanain, Fatema Ahmad, Sahinur Rahman Laskar, Sunaya Upadhyay, Vrunda N Sukhadia, Mucahid Kutlu, Shammur Absar Chowdhury, and Firoj Alam. 2025. NativQA: Multilingual culturally-aligned natural query for llms. In *Findings of the Association for Computational Linguistics: ACL 2025*, Vienna, Austria. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Kazi Samin, Masum Hasan, Madhusudan Basak, M. Sohel Rahman, and Rifat Shahriyar. 2020a. Not low-resource anymore: Aligner ensembling, batch filtering, and new datasets for Bengali-English machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2612–2623, Online. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Kazi Samin, Masum Hasan, Madhusudan Basak, M Sohel Rahman, and Rifat Shahriyar. 2020b. Not low-resource anymore: Aligner ensembling, batch filtering, and new datasets for bengali-english machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2612–2623.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022. Training compute-optimal large language models. *Preprint*, arXiv:2203.15556.

J Jaavid, Raj Dabre, M Aswanth, Jay Gala, Thanmay Jayakumar, Ratish Puduppully, and Anoop Kunchukuttan. 2024. Romansetu: Efficiently unlocking multilingual capabilities of large language models via romanization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 15593–15615.

Daeen Kabir, Minhajur Rahman Chowdhury Mahim, Sheikh Shafayat, Adnan Sadik, Arian Ahmed, Eunsu Kim, and Alice Oh. 2025. Bluck: A benchmark dataset for bengali linguistic understanding and cultural knowledge. *arXiv preprint arXiv:2505.21092*.

Mohsinul Kabir, Mohammed Saidul Islam, Md Tahmid Rahman Laskar, Mir Tafseer Nayeem, M Saiful Bari, and Enamul Hoque. 2024. BenLLM-Eval: A comprehensive evaluation into the potentials and pitfalls of large language models on bengali nlp. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2238–2252.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. IndicLLMSuite: A blueprint for creating pre-training and fine-tuning datasets for Indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879, Bangkok, Thailand. Association for Computational Linguistics.

Denis Kocetkov, Raymond Li, Loubna Ben allal, Jia LI, Chenghao Mou, Yacine Jernite, Margaret Mitchell, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Dzmitry Bahdanau, Leandro Von Werra, and Harm de Vries. 2023. The stack: 3 TB of permissively licensed source code. *Transactions on Machine Learning Research*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Langauge Technology (HLT-NAACL 2003)*, pages 48–54. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.

Wen Lai, Mohsen Mesgar, and Alexander Fraser. 2024. LLMs beyond English: Scaling the multilingual capability of LLMs with cross-lingual feedback. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8186–8213, Bangkok, Thailand. Association for Computational Linguistics.

Séamus Lankford, Haithem Afli, and Andy Way. 2022. Human evaluation of english–irish transformer-based nmt. *Information*, 13(7):309.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. BLOOM: A 176b-parameter open-access multilingual language model.

Aaron Levine, Connie Huang, Chenguang Wang, Eduardo Batista, Ewa Szymanska, Hongyi Ding, Hou Wei Chou, Jean-François Pessiot, Johanes Effendi, Justin Chiu, et al. 2024. Rakutenai-7b: Extending large language models for japanese. *arXiv e-prints*, pages arXiv–2403.

W Lian, B Goodson, E Pentland, et al. 2023. Openorca: An open dataset of gpt augmented flan reasoning traces. *arXiv preprint arXiv:2305.11206*.

Peiqin Lin, Shaoxiong Ji, Jörg Tiedemann, André FT Martins, and Hinrich Schütze. 2024. Mala-500: Massive language adaptation of large language models. *arXiv preprint arXiv:2401.13303*.

Siyang Liu, Naihao Deng, Sahand Sabour, Yilin Jia, Minlie Huang, and Rada Mihalcea. 2023a. Task-adaptive tokenization: Enhancing long-form text generation efficacy in mental health and beyond. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15264–15281.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. Gpteval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Minh-Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the seventeenth conference on computational natural language learning*, pages 104–113.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391.

Rabindra Nath Nandi, Mehadi Menon, Tareq Muntasir, Sagor Sarker, Quazi Sarwar Muhtaseem, Md. Tariqul Islam, Shammur Chowdhury, and Firoj Alam. 2023. Pseudo-labeling for domain-agnostic Bangla automatic speech recognition. In *Proceedings of the First Workshop on Bangla Language Processing (BLP-2023)*, pages 152–162, Singapore. Association for Computational Linguistics.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. 2024. Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237.

OpenAI. 2023. GPT-4 technical report. Technical report, OpenAI.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Hamza Alobeidli, Alessandro Cappelli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data only. *Advances in Neural Information Processing Systems*, 36:79155–79172.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 Indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Sagor Sarker. 2021. Bnlp: Natural language processing toolkit for bengali language. *arXiv preprint arXiv:2102.00405*.

Neha Sengupta, Sunil Kumar Sahu, Bokang Jia, Satheesh Katipomu, Haonan Li, Fajri Koto, Osama Mohammed Afzal, Samta Kamboj, Onkar Pandit, Rahul Pal, et al. 2023. Jais and Jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models. *arXiv preprint arXiv:2308.16149*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Sheikh Shafayat, H Hasan, Minhajur Mahim, Rifki Putri, James Thorne, and Alice Oh. 2024. BEnQA: A question answering benchmark for Bengali and English. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1158–1177, Bangkok, Thailand. Association for Computational Linguistics.

Md. Istiak Hossain Shihab, Md. Rakibul Hasan, Mahfuzur Rahman Emon, Syed Mobassir Hossen, Md. Nazmuddoha Ansary, Intesur Ahmed, Fazle Rabbi Rakib, Shahriar Elahi Dhruvo, Souhardya Saha Dip, Akib Hasan Pavel, Marsia Haque Meghla, Md. Rezwanul Haque, Sayma Sultana Chowdhury, Farig Sadeque, Tahsin Reasat, Ahmed Imtiaz Humayun, and Asif Shahriyar Sushmit. 2023. Badlad: A large multi-domain bengali document layout analysis dataset. *Preprint*, arXiv:2303.05325.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. In *Proceedings of the 62nd Annual Meeting*

of the Association for Computational Linguistics (Volume 1: Long Papers), pages 15725–15788, Bangkok, Thailand. Association for Computational Linguistics.

Tianxiang Sun, Junliang He, Xipeng Qiu, and Xuanjing Huang. 2022. BERTScore is unfair: On social bias in language model-based metrics for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3726–3739, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Nahida Akter Tanjila, Afrin Sultana Poushi, Sazid Abdullah Farhan, Abu Raihan Mostofa Kamal, Md. Azam Hossain, and Md. Hamjajul Ashmafee. 2025. Bengali ChartSumm: A benchmark dataset and study on feasibility of large language models on Bengali chart to text summarization. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 35–45, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. *arXiv preprint arXiv:2303.16199*.

Fanar Team, Ummar Abbas, Mohammad Shahmeer Ahmad, Firoj Alam, Enes Altinisik, Ehsannedin Asgari, Yazan Boshmaf, Sabri Boughorbel, Sanjay Chawla, Shammur Chowdhury, Fahim Dalvi, Kareem Darwish, Nadir Durrani, Mohamed Elfeky, Ahmed Elmagarmid, Mohamed Eltabakh, Masoomali Fatehkia, Anastasios Fragkopoulos, Maram Hasanain, Majd Hawasly, Mus'ab Husaini, Soon-Gyo Jung, Ji Kim Lucas, Walid Magdy, Safa Messaoud, Abubakr Mohamed, Tasnim Mohiuddin, Basel Mousi, Hamdy Mubarak, Ahmad Musleh, Zan Naeem, Mourad Ouzzani, Dorde Popovic, Amin Sadeghi, Husrev Taha Sencar, Mohammed Shinoy, Omar Sinan, Yifan Zhang, Ahmed Ali, Yassine El Kheir, Xiaosong Ma, and Chaoyi Ruan. 2025. Fanar: An arabic-centric multimodal generative ai platform.

Gemma Team. 2024. Gemma.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Atula Tejaswi, Nilesh Gupta, and Eunsol Choi. 2024. Exploring design choices for building language-specific llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10485–10500.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15894–15939, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Abdullah Khan Zehady, Safi Al Mamun, Naymul Islam, and Santu Karmaker. 2024. BongLLaMA: Llama for bangla language. *arXiv preprint arXiv:2410.21200*.

Qiang Zhang, Jason Naradowsky, and Yusuke Miyao. 2023. Ask an expert: Leveraging language models to

improve strategic reasoning in goal-oriented dialogue models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6665–6694.

Xiang Zhang, Juntai Cao, and Chenyu You. 2024. Counting ability of large language models and impact of tokenization. *arXiv preprint arXiv:2410.19730*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

A Pretraining Data

A.1 Tokens in *TituLLMs*

Table 4 presents the token distribution used for pretraining *TituLLMs*, which are collected from various sources. The largest portion, amounting to 18.80 billion tokens, originates from a deduplicated corpus combining Common Crawl, Books, and Sangraha data. Additionally, synthetic data generated for the model contributes 7.06 billion tokens, comprising translated (1.47b), romanized (3.87b), conversational (0.42b), and audio-transcribed (1.30b) text. The Sangraha dataset further adds 10.94 billion tokens, split between translated (4.26b) and romanized (6.68b) subsets. In total, the *TituLLMs* pretraining corpus consists of $\sim$ 37 billion tokens, incorporating both natural and synthetic data sources to enhance linguistic diversity and representation.

Data	**# Tokens (B)**
Common Crawl, Books, and Sangraha	18.80
Synthetic	
Translated	1.47
Romanized	3.87
Conversation	0.42
Audio Transcription	1.30
Sangraha	
Translated	4.26
Romanized	6.68
Total	**36.80**

Table 4: Token distribution for *TituLLMs* pretraining, including contributions from natural and synthetic data sources.

A.2 Rules for Data Filtering

For pretraining data filtering, we applied a set of carefully designed hand-crafted rules, which are outlined below.

Line Ending with Terminal Punctuation: Determines whether a line concludes with a terminal punctuation mark, including “.”, “!”, “?”, and “””. Helps assess sentence completeness and filter out incomplete or malformed content.

Line Word Numbers: Calculates the number of words in each line after normalization. Provides insights into whether text consists of single-word lines, short fragments, or full-length sentences.

Line Start with Bullet Points: Identifies lines starting with bullet points, including Unicode symbols like `\u2022`, `\u2023`, `\u25B6`, and others. Useful for recognizing and handling structured lists.

Line Numerical Character Fraction: Measures the proportion of numerical characters in each line. Helps identify lines dominated by numbers, such as statistics, mathematical expressions, or financial reports.

Is Adult URL: Flags documents originating from adult content URLs for filtering inappropriate or explicit content.

Document Languages Identification: Uses FastText (Grave et al., 2018) for language identification, extract language percentages, and filter documents where Bangla meets the specified threshold.

Document Sentence Count: Counts the number of sentences using BNLP Tokenizer (Sarker, 2021) for Bangla and NLTK Tokenizer [9]for English , and it helps to assess document length and complexity.

Document Word Count: Computes the total number of words after normalization. Provides an overall measure of document length and verbosity.

Document Mean Word Length: Calculates the average length of words after normalization. Longer words often indicate more sophisticated vocabulary.

Document Word to Symbol Ratio: Determines the ratio of symbols (“#”, “...”, “. . . ”) to words. A high ratio may indicate unconventional formatting or non-standard text.

Document Fraction End with Ellipsis: Computes the fraction of lines ending with an ellipsis (“...”, “. . . ”), which may suggest incomplete thoughts or trailing sentences.

[9]

Document Unique Word Fraction: Measures the fraction of unique words in a document, providing insight into vocabulary diversity and repetition.

Document Unigram Entropy: Calculates the entropy of the unigram distribution, measuring lexical variety using the formula:

$$\sum \left(-\frac{x}{\text{total}} \log \frac{x}{\text{total}} \right)$$

where x represents counts of unique words in the normalized content. Higher entropy suggests greater lexical diversity.

Document Stop Word Fraction: Determines the ratio of stop words (e.g., "the," "and," "is") to total words. A high ratio may indicate informal text, while a low ratio may suggest technical or keyword-dense content.

Fraction of Characters in Top N-Gram: Measures the proportion of characters within frequently occurring word n-grams. Helps assess text repetitiveness and structure.

Document Content Classification: Categorizes document content based on profanity, vulgarity, or toxicity to filter inappropriate material.

Document Bad Words Count: Counts offensive or inappropriate words, serving as a stricter filter for explicit content.

Document Bracket Ratio: Determines the ratio of all bracket types (e.g., "()", "[]", "{}") to total characters. Useful for detecting technical or structured text such as programming code, mathematical expressions, or legal documents.

By leveraging language-specific tokenizers and tools, we ensured that the evaluation framework effectively captured the characteristics and patterns unique to Bangla text, enabling robust filtering and alignment with intended use cases.

A.3 Rules for Cleaning OCR-Extracted Text

Figure 7 provides an overview of the book data collection process. We applied the following rules to filter the text from the OCR-extracted text.

- **Use of KenLM (Heafield, 2011):** KenLM is an efficient statistical language modeling toolkit commonly used for constructing n-gram language models. We trained a language model using high-quality text data, which enabled us to calculate word and sentence scores for the OCR-extracted text. A histogram of these scores was plotted, and thresholds were set based on the distribution to identify poorly recognized sections. The threshold for filtering low-quality text was set at 95%, meaning that only 95% of the text was retained.
- **Word and Sentence Count in Documents:** We calculated the average number of words and sentences per page in the collected books. A minimum threshold for these counts was established to help filter out books with low-quality text.
- **Percentage of Correct Bangla Words:** We compiled a list of common Bangla words and computed the percentage of these words in each book. A threshold was determined based on the overall distribution of Bangla word occurrences across the corpus.

Once these thresholds were established, they were applied across the entire dataset, resulting in the filtering of approximately 50% of the raw text data.

B Tokenizer Details

Table 5 presents the Tokens per Word (TPW) values for different variants of the Llama-3.2 model. The base Llama-3.2 model has the highest TPW at 7.8397, while the extended Llama-3.2-plus models, with varying context lengths (32K to 96K), exhibit progressively lower TPW values.

Model	TPW
`Llama-3.2`	7.8397
`Llama-3.2-plus-32K`	2.1346
`Llama-3.2-plus-48K`	1.9029
`Llama-3.2-plus-64K`	1.7946
`Llama-3.2-plus-80K`	1.7370
`Llama-3.2-plus-96K`	1.7034

Table 5: Tokens per word (TPW) for different Llama models.

C Translation Model (For pertaining data) Evaluation

Scalar Quality Metrics (SQM): We use Scalar Quality Metrics (SQM) (Lankford et al., 2022) to perform human-level evaluation of machine translation outputs in a structured and interpretable manner. SQM employs a 0 to 6 scale to assess the quality of each translated segment, considering both the semantic accuracy and grammatical correctness within the context of the document. This scale provides annotators with a simplified alternative to the traditional 0–100 WMT scoring method, while still

capturing nuanced quality differences. Key levels on the scale (0, 2, 4, 6) represent distinct degrees of translation quality—from complete nonsense (0) to perfect translations (6)—with intermediate scores (1, 3, 5) allowing for more precise evaluation. By using SQM, we ensure consistent and context-aware human judgments across different translation models.

To evaluate translation models, we have collected 100 sentences from the BBC News Dataset (Greene and Cunningham, 2006). We have divided the samples into five categories: Business, Entertainment, Politics, Tech, and Sport. We have 20 samples for each category. After that, we provided the translations from multiple models to two human evaluators (non-authors), and they scored the translations coming from each model.

D Expressive Semantic Translation (EST)

D.1 Expressive Semantic Translation

The EST method innovatively enhances neural machine translation by infusing expressiveness and contextual relevance through an iterative refinement process utilizing LLMs. The EST method encompasses several pivotal steps:

Initial Translation: A standard translation model M converts text from a source language L_1 into a preliminary translation t_0 in the target language L_2, which often lacks expressiveness and contextual depth.

Enhanced Linguistic Refinement: Employing multiple LLMs, the initial translation t_0 is refined into diverse candidate translations that exhibit greater naturalness and idiomatic correctness in L_2.

Quality Diversity: This phase synthesizes the best elements from the candidate translations through a prompt-based evaluation method, aiming to construct a translation that faithfully represents the nuances of L_2.

Candidate Ranking: Through model-based evaluations, candidates are assessed for contextual appropriateness and linguistic coherence, with the highest-scoring translation selected as the most suitable.

Final Selection: After multiple iterations, the optimal translation is selected from the top candidates based on rigorous evaluations, ensuring adherence to the linguistic and contextual standards of L_2.

These steps ensure that the final output not only meets the literal translation requirements but also enhances the translation quality by embodying the naturalness and contextual relevance, significantly surpassing traditional methods.

We evaluate EST method on a benchmarking test dataset by both well-known evaluation methods and LLM-based methods.

D.2 Evaluation

Data Selection: The dataset that has been utilized here for evaluation was taken from (Hasan et al., 2020b) where a customized sentence segmenter was used for Bangla and two novel methods, namely aligner ensembling and batch filtering, were used to develop a high-quality parallel corpus of Bangla and English with 2.75 million sentence pairings. The 1000 pairs that comprise up the test set of this data were created with extensive quality control and used in this assessment.

Evaluation metrics: We employ the BLEU Score (Papineni et al., 2002), SacreBLEU Score (Post, 2018), BERT Score (F1) (Sun et al., 2022), and an LLM-Based Evaluation.

LLM-based evaluation: This approach uses a large language model (GPT-4o) to assess translations qualitatively. The LLM is instructed by the given prompt to assess a Bangla translation against a reference text using the following criteria: accuracy (measures translation accuracy and semantic richness), fluency, readability, and faithfulness. A score between 1 and 10 is then assigned by the LLM, along with a rationale for each translation. Finally, an average score is computed for each translation method. LLM-based evaluation is more flexible and human-like than token-based methods since it may assess more semantic variations and fluency.

Result: Our comparison of the EST method against industry-standard models like Google Translation API and advanced systems like GPT-4o and Gemini highlights EST's superior performance. As shown in Table **??**, EST leads with remarkable scores in BLEU (**0.57**), SacreBLEU (**49.50**), and BERTScore (F1) (**0.93**). These metrics underscore EST's unparalleled accuracy and contextual richness, with a notable **20-point** lead in SacreBLEU over the closest competitor. These results affirm the efficacy of EST's iterative refinement in elevating translation quality.

E Conversation Data Generation Prompt

Our methodology for generating Bangla conversational texts involves two specialized roles: the Ju-

Translator	BLEU	SBLEU	BS (F1)	LLM Score
Google	0.33	29.00	0.91	**8.96**
GPT-4o	0.26	22.05	0.90	8.91
IndicTrans2	0.27	23.74	0.90	8.73
Gemini-1.5-pro	0.31	27.08	0.89	8.80
EST	**0.57**	**49.50**	**0.93**	8.95

Table 6: Comparison of different translation models.

nior Content Writer and the Senior Content Writer as highlighted in Figure 7. The Junior initiates dialogues based on culturally significant topics. The Senior meticulously reviews these texts to ensure grammatical precision and enhance quality. This structured approach enables replicable, high-standard conversation generation for NLP research.

Component	Description	Details
Input Text Snippet	Bangla text snippet provided to initiate the conversation.	The text is a prompt that provides a topic for conversation, such as "Rabindranath Tagore's contributions to Bengali art", sourced from educational and cultural databases like Wikipedia or Banglapedia. This serves as the basis for the conversation generation task.
Junior Content Writer	Agent tasked with generating the initial conversation.	**Role:** Content Creator **Goal:** To generate high-quality, coherent, and engaging conversational text in Bangla. **Backstory:** As an expert in Bangla language conversation generation, your responsibility includes initiating and maintaining a dialogue that is both interesting and relevant to the given topic. **Capabilities:** Produces verbose outputs to ensure detailed and extensive dialogue. No delegation is allowed. Employs a language learning model (LLM) without additional tools.
Senior Content Writer	Agent tasked with reviewing and refining the conversation.	**Role:** Examiner **Goal:** To edit and enhance the quality of Bangla conversational text, ensuring it meets high standards of grammar and coherence. **Backstory:** You review the conversation generated by the Junior Content Writer, focusing on grammatical accuracy, linguistic quality, and content relevance, making necessary revisions to uphold quality standards. **Capabilities:** Provides verbose feedback and detailed edits, uses a language model (LLM) for text correction and improvement, and operates independently without delegation.

Table 7: Detailed roles and tasks of agents for generating Bangla conversational text.

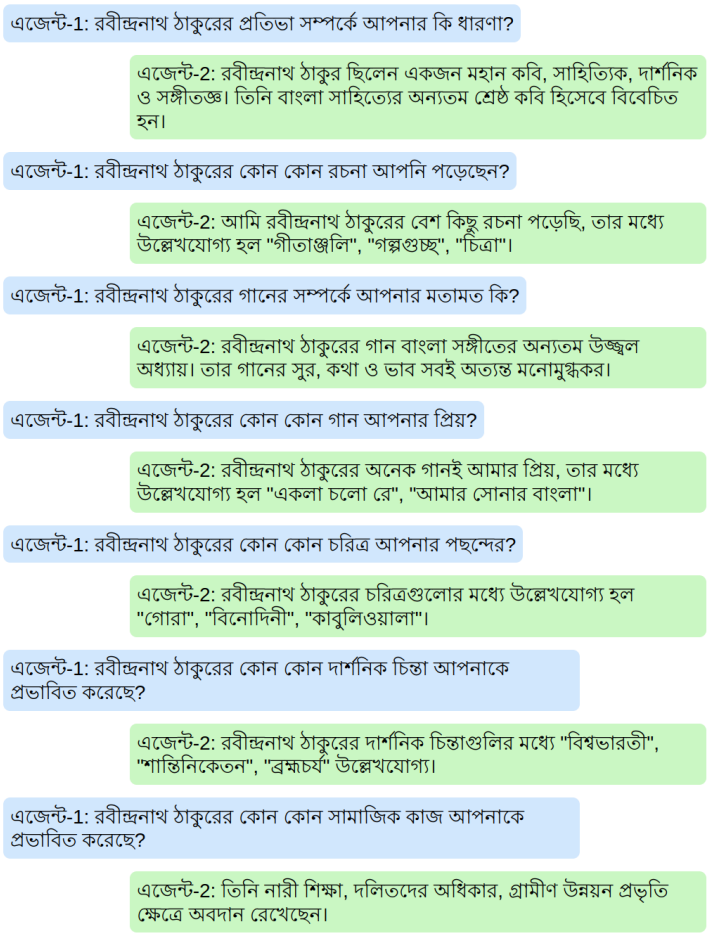

Figure 7: Conversation example between two agents on Rabindranath Tagore's contributions to Bangla art.