ChakmaNMT: Machine Translation for a Low-Resource and Endangered Language via Transliteration

Anonymous ACL submission

Abstract

We present the first systematic study of machine translation for Chakma, an endangered and extremely low-resource Indo-Aryan language, with the goal of supporting language access and preservation. We introduce a new Chakma–Bangla parallel and monolingual dataset, along with a trilingual Chakma–Bangla–English benchmark for evaluation. To address script mismatch and data scarcity, we propose a character-level transliteration framework that exploits the close orthographic and phonological relationship between Chakma and Bangla, preserving semantic content while enabling effective transfer from Bangla and multilingual pretrained models. We benchmark from-scratch MT, fine-tuned pretrained models, and large language models via in-context learning. Results show that transliteration is essential and that fine-tuning and in-context learning substantially outperform from-scratch baselines, with strong asymmetry across translation directions.

Src.	𑄌𑄋𑄧𑄢𑄴𑄝𑄳𑄢 𑄟𑄪𑄨𑄛𑄳𑄇 𑄌𑄪𑄢 𑄝𑄪𑄢 𑄋𑄬𑄢𑄴𑄌𑄪𑄚𑄳 𑄝𑄬𑄢𑄴𑄟𑄪𑄛𑄳𑄟𑄪𑄛 𑄝𑄬𑄋𑄬𑄢 𑄎𑄬𑄌𑄬𑄚𑄳 𑄌𑄬𑄢𑄴𑄚𑄪𑄢 ।
Ref.	দেশকে সামনের দিকে এগিয়ে নিয়ে যেতে বেসরকারি সেক্টর খুবই গুরুত্বপূর্ণ । (To move the country forward, private sector is very important.)
NMT	দেশটিকে সামনে এগিয়ে নিয়ে যাওয়ার জন্য বেসিরকারি সেক্টরের যথেষ্ট ভূমিকা রয়েছে । (To move the country forward, there is a significant importance of the private sector.)
ICL	দেশগুলোকে বিভিন্নভাবে আরও বেশি করে বেসরকারি খাতে যুক্ত করতে হবে যেখানে সম্ভব। (Countries need to engage the private sector more and more in various ways where possible.)

Figure 1: Illustrative Chakma→Bangla translation example comparing our two best-performing approaches: fine-tuned NMT (BanglaT5) and in-context learning (GPT with random 400 examples). Despite similar automatic scores, the outputs differ in lexical choice and interpretation.

1 Introduction

The Chakma language is spoken by the indigenous Chakma people across Bangladesh, the easternmost regions of India, and western Myanmar, and belongs to the Indo-Aryan language family (Mohsin, 2013). It is spoken by over 700,000 people across the region (censusindia, 2011; Statistics, 2023). Despite this population, Chakma remains predominantly oral, with limited use of its writing system, leaving the language critically underresourced in digital form. As noted by Saikia and Haokip (2023), Chakma is classified as "Definitely Endangered," and continued language loss poses risks to cultural identity and community continuity. Although revitalization efforts exist-such as pre-primary materials produced by the National Curriculum and Textbook Board of Bangladesh (NCT, 2024) and limited school textbooks in India (CAD, 2024)-they remain largely confined to education, while everyday communication and public discourse increasingly rely on dominant regional languages such as Bangla, underscoring the lack of computational support for Chakma in broader communication settings.

Compared to higher-resource Indo-Aryan languages such as Bangla and Hindi, Chakma has received very limited attention in NLP research. Existing work is largely restricted to character recognition (Podder et al., 2023) and speech language identification (Pratap et al., 2023). In contrast, while state-of-the-art commercial large language models (e.g., GPT and Grok) can recognize Chakma script, they often fail to generate semantically faithful Chakma sentences, limiting their reliability in this low-resource setting (see Section 7). As a result, foundational text-based capabilities-most notably machine translation (MT), which is criti-

cal for cross-lingual communication and access to public resources-remain largely unexplored.

Motivated by this gap, we present the first systematic study of machine translation for Chakma, an extremely low-resource and endangered language. We investigate how different modeling paradigms, including classical machine translation, neural models, and large language models via in-context learning, behave under severe data scarcity and cross-script conditions. Our goal is to establish practical baselines for Chakma MT while highlighting challenges that arise in extremely low-resource and non-standardized languages. Figure 1 illustrates qualitative differences between our best-performing systems. A fine-tuned BanglaT5 model produces accurate translations with limited parallel data. A GPT-based in-context learning approach, using only 400 demonstration examples, generates outputs that remain semantically faithful despite minimal supervision.

Our contributions are summarized as follows:

(1) **First Chakma–Bangla MT resources.** We release the first Chakma–Bangla parallel corpus with 15,021 sentence pairs, a large Chakma monolingual corpus with 42,783 sentences, and a curated trilingual Chakma–Bangla–English benchmark with 600 evaluation sentences. These resources provide a foundation for machine translation and downstream NLP research on Chakma.

(2) **Script-bridging transliteration framework.** We propose a simple, rule-based character-level transliteration system that leverages Chakma–Bangla script similarity to provide near one-to-one mappings, preserving semantic content while bridging script differences. This enables effective cross-script transfer and allows pretrained and large language models to be applied in this extremely low-resource setting.

(3) **Comprehensive benchmarking in an extremely low-resource setting.** We systematically benchmark statistical and neural MT, fine-tuned pretrained models (e.g., BanglaT5, mT5), and GPT-based in-context learning, establishing strong and robust baselines for Chakma-Bangla translation.

(4) **Analysis of orthographic variability.** We analyze orthographic inconsistencies in Chakma arising from non-standardized spelling and script usage. We show that this variability substantially affects automatic evaluation and MT performance, with BLEU underestimating translation quality due to multiple valid spellings.

2 Related Works

Machine translation research has historically focused on high-resource language pairs, where large parallel corpora were readily available (Koehn and Knowles, 2017). Early work was dominated by statistical machine translation systems trained on millions of sentence pairs for high-resource language pairs (Koehn et al., 2003). With the advent of neural machine translation, attention-based encoder–decoder models further reinforced this reliance on abundant parallel data (Bahdanau et al., 2014). As a result, languages with scarce digital resources have received comparatively less attention and continue to face substantial limitations in existing MT systems.

More recent work in low-resource NMT has explored a range of strategies to mitigate data scarcity, including semi-supervised learning with monolingual data (Gulcehre et al., 2015), back-translation (Sennrich et al., 2016), multilingual neural machine translation for cross-lingual transfer (Kocmi and Bojar, 2018), and transliteration for closely related languages with different scripts (Durrani et al., 2010), particularly for Asian and Indigenous languages (Riza et al., 2016). In extremely low-resource settings, prior work reports very low translation quality overall (often below 10 BLEU) (Guzmán et al., 2019), with several MT approaches yielding BLEU scores in the low single digits or around 1–2 under out-of-domain evaluation, highlighting the difficulty of generalization (Zhang et al., 2020).

More recently, large language models have introduced in-context learning (ICL) as an alternative to fine-tuning for low-resource translation (Brown et al., 2020). While promising, ICL performance is direction-dependent and varies across language pairs, often favoring high-resource target languages (Brown et al., 2020). These limitations motivate us to systematically study the effectiveness of ICL in extremely low-resource and cross-script translation settings.

3 ChakmaNMT Dataset

Table 1 summarizes the parallel, monolingual, and evaluation data collected in this work, with details discussed below.

3.1 Parallel Corpus

Parallel Documents After extensive searching, we identified two publicly available documents that

Category	Source	Samples	Avg #Tok	Total
Parallel	UN-Disabilities (BN-CCP)	610	16.86	15,021
	UN-Child Rights (BN-CCP)	291	37.66	
	Dictionary (word pairs)	5,473	1.14	
	Crowdsourced (BN-EN-CCP)	3,444	4.51	
	Expert translations (BN-EN-CCP)	5,203	3.60	
Monolingual	Local Chakma sources (CCP)	42,783	5.81	42,783
Evaluation	RisingNews Benchmark Extension (BN-EN-CCP)	600	14.8	600

Table 1: Overview of the Chakma–Bangla datasets introduced in this work, including data sources, sample counts, and average sequence length (Avg #Tok) measured in space-separated Chakma (CCP) tokens; BN and EN denote Bangla and English, respectively.

contain aligned Bangla(BN) and Chakma(CCP) translations: *UN Convention on the Rights of Persons with Disabilities* (UnitedNation) and *UN Convention on the Rights of the Child* (resolution 44/25, 1989). The Bangla versions of these documents were available only as scanned PDF files containing images of the original printed documents. We applied Tesseract OCR[1] to extract Bangla text from these scans.

Automatic sentence alignment methods such as Hualign (Varga et al., 2005) were not applicable in our setting due to the lack of a sufficiently rich Chakma lexicon. Consequently, all sentence alignments were performed manually. This process resulted in 610 and 291 CCP-BN sentence pairs from the two documents, respectively. In addition, we incorporated word-level translation pairs from the only available Chakma dictionary, provided to us in JSON format.[2] This yielded 5,473 additional parallel samples.

Manual Translation by Experts To obtain high-quality sentence-level translations, we organized a manual data collection effort involving native Chakma speakers with strong literacy skills. We prepared paper-based forms containing 10,000 Bangla sentences randomly sampled from the BN-EN corpus of Hasan et al. (2020). A three-day voluntary translation program was conducted in Dighinala, Khagrachari (Bangladesh). Between 7 and 10 proficient Chakma speakers participated each day.

All collected translations were subsequently reviewed and validated by senior linguistic experts to ensure accuracy and consistency. After filtering and quality control, this process produced a final set of 5,203 high-quality CCP-BN-EN parallel sentence triples. Further details of this collection process are provided in Appendix A.1.

Manual Translation via Crowdsourcing To further expand the dataset, we collected additional translations through a crowdsourcing approach involving non-expert Chakma speakers. We developed a web-based platform that displayed Bangla sentences and allowed users to submit their Chakma translations. The platform link was distributed through social media channels.

The source sentences primarily consisted of common conversational expressions collected from publicly available resources[3]. These resources already include English translations. Since most contributors were unfamiliar with the Chakma script, they were instructed to write Chakma using Bangla characters. We later converted these submissions into Chakma script using a custom transliteration system (See Section 4.2) developed for CCP-BN conversion. After manual verification and filtering by Chakma language experts, this effort yielded 3,444 additional CCP-BN-EN sentence triples.

Overall, the parallel data collection process resulted in 15,021 BN-CCP parallel sentence pairs, of which 8,647 include aligned English translations.

3.2 Monolingual Data

Figure 2 shows the distribution of the collected Chakma monolingual data by content type. We collected a substantial amount of Chakma monolingual data relative to the available parallel resources. Due to the scarcity of digitally available

[1] https://github.com/tesseract-ocr/tesseract

[2] The dictionary data were provided directly by the dictionary's owner for research use.

[3] https://www.learnenglishfrombangla.com/2021/07/easily-learn-english-in-bangla-beginner.html, https://www.omniglot.com/language/phrases/bengali.php, and https://en.wikibooks.org/wiki/Bengali/Common_phrases

Chakma texts,[4] we conducted in-person visits to Chakma language scholars to obtain soft copies of Chakma script materials. These materials primarily consist of poems, articles, short stories, and a small number of national-level textbooks. In addition, we collected Indian Chakma textbooks and a Chakma folktale mobile application, and we reused the Chakma dictionary introduced in the parallel data collection, which contains numerous high-quality example sentences accompanying lexical entries and is therefore well suited for monolingual data extraction.

To process these sources, all materials were first transcribed into separate `.docx` files, which preserved the original Chakma fonts used in the documents. However, these fonts were encoded in various ASCII-based formats, each with distinct character mappings. To address this, we developed a conversion program that maps all source fonts to a unified Unicode font, RebangUni,[5] the first UTF-8–compliant font for the Chakma language. This enabled consistent normalization across heterogeneous sources.

Finally, we applied a simple rule-based segmentation procedure, splitting text at sentence boundaries defined by three punctuation markers: ?', !', and 'l' (full stop). After preprocessing and normalization, we obtained a total of 42,783 Chakma monolingual samples. Tables 3 and 4 provide detailed descriptions of the monolingual data sources, while the font conversion code is available in our GitHub repository[6] and the list of supported ASCII fonts is provided in Appendix Table 5.

3.3 Evaluation Data

To evaluate our models, we constructed a carefully curated benchmark dataset (see Table 1). We randomly selected 500 Bangla–English sentence pairs from the RisingNews Benchmark dataset, which consists of online news articles, introduced by Hasan et al. (2020). The RisingNews dataset was preprocessed and filtered following the methodology of Guzmán et al. (2019), making it a high-quality and widely used evaluation resource. Since the dataset already contains bilingual sentence pairs, translating these sentences into Chakma enables the construction of a trilingual benchmark spanning Bangla, English, and Chakma. We asked three Chakma language researchers, who had not participated in the parallel data collection, to independently translate these sentences into Chakma. Each annotator translated 200 sentences, with an overlap of 50 sentences shared across all three annotators. The shared subset was included to allow analysis of translation variability and orthographic inconsistency across gold references, which we further discuss in Section 7. In total, this process resulted in 600 evaluation samples, which we refer to as the *RisingNewsChakma* benchmark. This benchmark is out-of-domain with respect to our training data and is used exclusively for evaluation.

4 Machine Translation Approaches

Machine Translation Task This task is formulated as a sequence-to-sequence learning problem at the sentence level. Given a source-language sentence, the model generates a target-language sentence token by token. The objective is to learn this mapping from extremely limited parallel data. We study this setting for machine translation between Chakma and Bangla.

We compare three complementary approaches to establish strong baselines and understand what works best for this language pair. Our methods differ primarily in how they leverage prior knowledge and handle the script mismatch between Chakma and Bangla. First, we train conventional MT systems from scratch using only our collected parallel data, which are directly limited by data scarcity. Second, we fine-tune pretrained sequence-to-sequence models by transferring knowledge from Bangla via script-bridging transliteration and monolingual augmentation. Third, we evaluate large language models using few-shot in-context learning, adapting them to Chakma translation without any parameter updates.

4.1 From-Scratch MT Baselines

We train both statistical and neural baselines from scratch on our parallel corpus. We use phrase-based SMT (Koehn et al., 2003) as a classical baseline that remains competitive in low-resource scenarios (Koehn and Knowles, 2017). We also train a GRU-based RNN with attention (Bahdanau et al., 2014; Luong et al., 2015) as a lightweight neural model. Finally, we train a Transformer (Vaswani et al., 2017) as a stronger but more data-demanding neural baseline.

[4] Chakma digital texts and scripts are rarely available online, and most existing materials are accessible only through local scholars or printed sources.

[5] https://github.com/Bivuti/RibengUni

[6] anonymousfornow.com

4.2 Script-Bridging Transliteration

To enable the use of pretrained sequence-to-sequence and large language models, we develop a rule-based, character-level transliteration system that bridges the Chakma and Bangla scripts in a near one-to-one manner.[7] The system preserves phonetic and lexical content while mapping Chakma text into the Bangla Unicode range, allowing Chakma data to be directly processed by Bangla-pretrained models. This transliteration step serves as a core foundation for both fine-tuning pretrained models and in-context learning experiments.

Exploiting a unique relationship between Chakma and Bangla. Chakma and Bangla exhibit an unusually high degree of phonetic and orthographic similarity among Indo-Aryan languages, making transliteration a natural and low-effort strategy to bridge script mismatch and enable effective transfer from Bangla-pretrained models. Beyond script, the languages also share a largely similar subject–object–verb (SOV) word order, which further supports cross-lingual transfer. At the same time, Chakma has systematic morphosyntactic differences (e.g., placing negation before the verb), which may introduce local reordering effects beyond script-level variation.

The transliteration is largely based on straightforward one-to-one character mappings in both directions, with a small number of deterministic, phonetics-based normalizations only where exact script-level equivalence is unavailable. The system prioritizes content preservation over strict character reversibility: round-trip transliteration may introduce minor surface-level variation, but does not result in semantic or lexical information loss (Section 7). In practice, transliteration is used exclusively as preprocessing and postprocessing: if Chakma is the source, input is transliterated into Bangla before translation; if Chakma is the target, Bangla-script model output is transliterated back into Chakma for evaluation. All pretrained sequence-to-sequence models described in the following subsection are trained and evaluated using this transliterated input–output representation.

We provide full mapping statistics, handling of non-direct characters, and a summary of all non-direct rules (Figure 3) in Appendix A.3.

[7]The transliteration code is publicly available on GitHub (anonymous).

4.3 Fine-Tuning Pretrained Sequence-to-Sequence Models

We fine-tune pretrained text-to-text models on transliterated Chakma–Bangla parallel data. We experiment with BanglaT5 (Bhattacharjee et al., 2023), mT5-small (Xue et al., 2021), and mBART (Liu et al., 2020). These models allow us to transfer linguistic knowledge from Bangla and multilingual pretraining into the Chakma setting.

We further evaluate two data-centric extensions to improve robustness under scarcity. We apply iterative back-translation (IBT) (Hoang et al., 2018) to generate synthetic parallel data from monolingual corpora. In IBT, we start with the forward direction CCP→BN trained on the original parallel data, the backward direction (BN→CCP) is trained with synthetic data, and in later iterations both directional models are trained with additional synthetic data. We also evaluate multilingual training (MNMT) following Johnson et al. (2017). For MNMT, we add 10k Bangla–English sentence pairs from Hasan et al. (2020) to the training data to encourage cross-lingual transfer across Bangla, Chakma, and English.

4.4 Large Language Models via In-Context Learning

We evaluate few-shot in-context learning as an alternative to fine-tuning using state-of-the-art large language models. Specifically, we test GPT-4.1, GPT-4.1-mini, and GPT-o4-mini with a fixed prompting template and a limited number of demonstration translation pairs. This setting assesses their ability to perform Chakma translation using only in-context examples. The full prompting template and example format are shown in Figure 4.

Example Selection Demonstration examples are selected using two retrieval strategies: uniform random sampling from the parallel data and character-level n-gram similarity with the input sentence. We use character-level matching to handle orthographic variation in Chakma, where multiple valid spellings make word-level retrieval unreliable. Comparing these strategies allows us to evaluate whether demonstration relevance, beyond the number of examples, improves in-context translation quality.

5 Experimental Setup

Data Splits and Evaluation We split the parallel Chakma–Bangla corpus into training and devel-

opment sets.[8] The training set contains 12,016 sentence pairs and the development set contains 3,005 sentence pairs. We evaluate all models on the RisingNewsChakma benchmark, which is out-of-domain with respect to the training data. We report BLEU (Post, 2018) and chrF (Popović, 2015) as our primary automatic metrics; following common practice, BLEU is used for model selection on the development set, and chrF is reported alongside BLEU for all experimental results.

From-Scratch SMT and NMT We use the Moses toolkit[9] for phrase-based SMT and PyTorch for neural models. All neural models are trained on Google Colab using NVIDIA V100/A100 GPUs[10]. Data preprocessing follows the normalization[11] scheme of Hasan et al. (2020) with minor language-specific adjustments. We apply SentencePiece (Kudo and Richardson, 2018) for tokenization across SMT and NMT systems. Decoding uses beam search with width 5 and a maximum sequence length of 128 tokens. We train the GRU-based and Transformer models described in Section 4.1.

Fine-Tuning and In-Context Learning We fine-tune BanglaT5, mT5-small, and mBART for Chakma–Bangla translation. For iterative back-translation, we use the full Chakma monolingual corpus and 50k Bangla monolingual sentences.

We also evaluate large language models using few-shot in-context learning without parameter updates, including GPT-4.1, GPT-4.1-mini, and GPT-o4-mini. We use default decoding settings with temperature set to 1 due to budget constraints. Each prompt includes between 100 and 400 example translation pairs and translates 20 input sentences, selected as a practical compromise between prompt utilization and computational cost. Demonstration examples are retrieved from the training split of the parallel corpus and selected either randomly or via character-level n-gram similarity, with $n = 6$ fixed for stability.

We additionally conduct ablation experiments by (i) removing transliteration for fine-tuned and in-context models, and (ii) evaluating a zero-shot in-context learning configuration without demonstrations.

Additional experimental details necessary for replication, including normalization, training hyperparameters, model architectures and initialization, multilingual data formatting and oversampling, in-context learning prompt construction, and multiple runs and randomness, are provided in Appendix A.4.

6 Results

Transliteration is essential for effective modeling Transliteration is a prerequisite for leveraging pretrained models and yields substantial gains. As shown in Table 6, removing transliteration collapses both BLEU and chrF to near-zero across fine-tuning and in-context learning. Reintroducing transliteration restores usable scores in both translation directions, showing that script conversion is essential. The parallel drops in BLEU and chrF indicate a failure at the character level rather than a tokenization issue.

In-context learning is the most effective approach, but directionally asymmetric With 400 examples, ICL achieves strong CCP→BN performance and is competitive with the best fine-tuned systems, where BanglaT5 consistently outperforms mT5 (Table 2 and Table 8). In contrast, BN→CCP performance remains low (around 1–2 BLEU) even with the best prompts, while fine-tuned models continue to perform better. This reveals a clear directional asymmetry: ICL is substantially more effective when translating into Bangla than into Chakma. chrF mirrors this pattern, showing much higher character-level overlap for CCP→BN than for BN→CCP.

From-scratch models fail to generalize under extreme data scarcity From-scratch SMT, RNN, and Transformer models achieve modest dev performance but collapse on the test set, with both BLEU and chrF dropping to near-zero levels (Table 2). This degradation holds in both translation directions and indicates a failure to generalize under extreme data scarcity. The parallel collapse in BLEU and chrF further suggests a strong domain mismatch with the RisingNewsChakma benchmark, where models fail to recover even partial character-level matches.

Back-translation improves performance in most settings Back-translation improves performance in most settings, with consistent gains across both BLEU and chrF. The improvements are strongest

[8] The dataset is anonymous for review and will be released publicly on HuggingFace upon publication.

[9] https://www2.statmt.org/moses/

[10] https://colab.research.google.com/

[11] https://github.com/anonymous_for_now

System	BN→CCP				CCP→BN			
	Dev		Test		Dev		Test	
	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
From-scratch trained								
SMT	04.6 ± −−	**26.3** ± −−	00.1 ± −−	20.2 ± −−	03.2 ± −−	24.9 ± −−	00.1 ± −−	**20.4** ± −−
RNN	**05.3** ± 0.91	22.2 ± 3.71	00.1 ± 0.03	10.4 ± 2.45	**08.0** ± 2.51	20.1 ± 4.37	00.1 ± 0.07	08.5 ± 2.43
Transformer	01.6 ± 0.03	25.8 ± 0.21	**00.2** ± 0.04	**23.2** ± 0.30	03.0 ± 0.12	**27.2** ± 0.28	**00.4** ± 0.05	20.0 ± 0.51
Fine-tuned with tranliteration enabled								
mBART	03.5 ± 0.35	20.0 ± 1.45	00.2 ± 0.03	11.1 ± 1.09	10.4 ± 6.74	22.6 ± 9.80	00.8 ± 0.51	12.8 ± 3.49
mT5-small	08.5 ± 0.11	32.2 ± 0.20	02.0 ± 0.04	25.8 ± 0.12	14.1 ± 0.33	33.9 ± 0.35	04.6 ± 0.07	27.2 ± 0.25
+IBT-1it	08.4 ± 0.03	33.2 ± 0.04	02.5 ± 0.02	29.4 ± 0.09	14.1 ± 0.33	33.9 ± 0.35	04.6 ± 0.07	27.2 ± 0.25
+IBT-2it	08.3 ± 0.09	33.9 ± 0.14	**02.7** ± 0.09	30.1 ± 0.09	15.1 ± 0.11	37.4 ± 0.14	07.8 ± 0.10	37.8 ± 0.20
+MNMT	06.1 ± 0.14	27.5 ± 0.35	01.5 ± 0.05	24.1 ± 0.73	09.2 ± 0.28	28.1 ± 0.30	03.0 ± 0.10	23.5 ± 0.34
BanglaT5	**10.9** ± 0.06	36.3 ± 0.22	**02.7** ± 0.05	**30.6** ± 0.21	**24.4** ± 0.11	44.6 ± 0.08	11.7 ± 0.19	37.9 ± 0.09
+IBT-1it	10.4 ± 0.19	36.5 ± 0.11	02.2 ± 0.06	27.7 ± 0.21	**24.4** ± 0.11	44.6 ± 0.08	11.7 ± 0.19	37.9 ± 0.09
+IBT-2it	10.5 ± 0.18	**36.8** ± 0.07	02.5 ± 0.18	28.5 ± 0.35	24.2 ± 0.11	**46.8** ± 0.06	**13.9** ± 0.20	**46.3** ± 0.15
+MNMT	08.2 ± 0.26	32.1 ± 0.22	02.4 ± 0.11	29.2 ± 0.32	20.9 ± 0.62	40.1 ± 0.39	12.8 ± 0.12	38.5 ± 0.61
In-context Learning with tranliteration enabled								
GPT-4.1-mini(R)	-	-	01.1 ± 0.06	28.4 ± 0.19	-	-	10.9 ± 0.19	40.0 ± 0.58
GPT-4.1(R)	-	-	**01.6** ± 0.13	30.4 ± 0.12	-	-	16.5 ± 0.12	48.2 ± 0.35
GPT-o4-mini(R)	-	-	01.5 ± 0.03	29.7 ± 0.03	-	-	12.5 ± 0.47	42.9 ± 0.08
GPT-4.1-mini(N)	-	-	01.2 ± 0.01	28.7 ± 0.25	-	-	10.5 ± 0.54	40.4 ± 0.47
GPT-4.1(N)	-	-	01.5 ± 0.10	**31.3** ± 0.53	-	-	**17.8** ± 0.12	**49.6** ± 0.12
GPT-o4-mini(N)	-	-	01.2 ± 0.07	29.7 ± 0.14	-	-	12.2 ± 0.27	42.7 ± 0.48

Table 2: Translation performance on BN↔CCP across development and test sets. We report mean ± standard deviation for BLEU and chrF. Results are shown for from-scratch models, fine-tuned pretrained models (including iterative back-translation(IBT), up to two iterations, and multilingual fine-tunin(MNMT), and GPT-based in-context learning (ICL) with 400 examples using random(R) or n-gram(N) similarity-based sampling. Overall, GPT-based ICL achieves the strongest performance for CCP→BN, while BanglaT5 yields the highest BLEU for BN→CCP.

for CCP→BN: for BanglaT5, two iterations increase test BLEU from 11.7 to 13.9 and chrF from 37.9 to 46.3, while mT5-small improves BN→CCP from 2.0 to 2.7 (Table 2). Gains in BN→CCP are smaller or mixed, although the forward model benefits from additional synthetic data in later IBT steps. Notably, CCP→BN scores for BanglaT5 and mT5 remain unchanged after the first IBT iteration, since the backward model is initially trained on the same parallel data. Overall, chrF closely mirrors BLEU, indicating that back-translation improves character-level fidelity rather than only n-gram overlap.

Bilingual fine-tuning outperforms multilingual training in our setting Multilingual training underperforms bilingual fine-tuning across both translation directions. For both BanglaT5 and mT5-small, adding MNMT reduces test performance on both BLEU and chrF relative to bilingual fine-tuning (Table 2). This indicates that the additional English data introduces noise that outweighs any cross-lingual transfer benefits in this low-resource setting. Table 9 further supports this finding, as EN↔CCP results are substantially worse than BN↔CCP on both metrics, showing degradation even at the character level.

ICL benefits from relevant demonstrations and careful scaling N-gram similarity–based selection consistently matches or slightly outperforms random sampling, particularly for CCP→BN (Table 8). Increasing the number of demonstrations improves performance on both BLEU and chrF up to a point, after which gains plateau and become non-monotonic. This shows that example relevance matters more than sheer quantity, with chrF mirroring BLEU and indicating improved character-level fidelity rather than just word overlap.

Performance-cost trade-offs between ICL and fine-tuning ICL achieves strong CCP→BN performance on both BLEU and chrF with minimal data, but requires large commercial models and costly prompts. Fine-tuning is cheaper and more stable, particularly for BN→CCP where ICL underperforms on both metrics. As a result, the preferred

approach depends on whether one prioritizes peak performance under data scarcity or long-term deployment cost, with ICL's gains concentrated in CCP→BN and fine-tuning remaining more reliable for BN→CCP.

7 Qualitative Analysis

BN→CCP translation is substantially harder than CCP→BN Across all systems, BN→CCP performance is substantially lower than CCP→BN on both BLEU and chrF (Table 2). Even the best BN→CCP results reach only about 2–3 BLEU, while CCP→BN attains 13–18 depending on the method. This asymmetry likely arises because pretrained models encode Bangla more effectively, making translation into Bangla easier than into Chakma. The same gap in chrF confirms that this is a genuine character-level difficulty rather than a BLEU-specific artifact. Figure 5 provides representative BN→CCP outputs across model families.

BLEU underestimates quality due to lexical and orthographic variation of Chakma Language We observe a large divergence between BLEU and chrF that reflects lexical and orthographic variation rather than semantic errors. BLEU is particularly sensitive to re-transliterated Chakma outputs, where spelling variation introduced by script conversion lowers n-gram overlap without degrading meaning. This is evident in inter-annotator agreement on 50 shared benchmark sentences (Section 3.3), which is only 4.48 BLEU but 38.82 chrF, indicating stable character overlap despite differing word forms. Figure 6 further illustrates multiple valid spellings for common words, strongly penalizing BLEU. These patterns, driven by the lack of standardized Chakma orthography, motivate treating chrF as a co-primary metric alongside BLEU.

Our transliteration preserves content but is not character-exact Round-trip evaluation shows that transliteration largely preserves the input, although it is not perfectly character-faithful. After one cycle, scores remain strong (41.55 BLEU / 79.32 chrF for BN→CCP→BN and 38.37 BLEU / 79.69 chrF for CCP→BN→CCP), indicating only minor character-level drift. After the second cycle, scores reach near-ceiling levels (97.6–100 BLEU and 99.5–100 chrF; Table 10), reflecting stabilization once non-bijective mappings are resolved. Overall, residual differences are best explained by a small set of nearest-character (phonetic) substitutions for symbols without exact cross-script counterparts—surface variations that preserve pronunciation and meaning rather than causing substantive information loss. This interpretation is consistent with downstream MT results (Table 2), where strong systems maintain relatively high chrF despite lower BLEU, while removing transliteration causes both metrics to collapse (Table 6).

Zero-shot ICL ablation highlights the need for demonstrations As an ablation, we evaluate zero-shot ICL without any demonstrations. For BN→CCP, BLEU remains below 1 across models (Table 11), indicating that zero-shot ICL is ineffective in this direction. CCP→BN performs better even without examples, but still lags behind few-shot ICL. These results confirm that explicit demonstrations are essential for generating Chakma outputs in this extremely low-resource setting. chrF is similarly low in zero-shot BN→CCP, underscoring that models fail to recover even partial character overlaps without examples.

LLM variants show different effectiveness under ICL Under identical in-context learning (ICL) setups, GPT-4.1 achieves the strongest overall performance, while GPT-o4-mini consistently outperforms GPT-4.1-mini in both BLEU and chrF (Table 8). As these models differ in architecture, capacity, and intended design, we do not attribute the observed differences to any single factor. Instead, we report them as an empirical comparison of LLM variants under the same ICL conditions. The chrF ranking matches BLEU, suggesting that model differences affect both token-level and character-level fidelity.

8 Conclusion

This work presents a foundational study on machine translation for Chakma, an extremely low-resource and endangered language. We introduce new datasets and a transliteration-based framework that enables effective use of pretrained models and large language models. Results show that leveraging pretrained models and related high-resource languages substantially outperforms training from scratch, while challenges such as translation asymmetry and orthographic variation remain. Overall, this work establishes strong baselines and a practical foundation for future NLP research on endangered languages.

Ethics

This work involves data collection for an endangered and low-resource language with the goal of supporting language preservation and accessibility. All human-generated data were collected with informed consent from contributors, who voluntarily participated and expressed support for this research. No personally identifiable information was collected, and we do not anticipate any significant risks or harms resulting from this work.

Limitations

This work is constrained by the extremely low-resource nature of the Chakma language, which limits the size and diversity of available training data and leads to relatively low automatic evaluation scores, a common challenge in low-resource machine translation. Our rule-based transliteration framework enables effective cross-script transfer and preserves phonetic and lexical content, but relies on manually designed mappings and is not strictly character-bijective, resulting in minor surface-level variation for a small number of script-specific distinctions without affecting meaning. Automatic metrics such as BLEU may further underestimate translation quality due to orthographic variation and multiple valid spellings in Chakma. Ultimately, the primary bottleneck remains data scarcity: future work could benefit substantially from automated web-based data crawling and collection systems to expand Chakma textual resources, as well as from more data-driven transliteration and translation approaches tailored to extremely low-resource and non-standardized languages.

References

2024. Chakma textbooks by chakma autonomous district council(cadc). CADC.

2024. Textbook of small ethenic group(chakma) for pre-primary. NCTB.

Dzmitry Bahdanau, Kyunghyun Cho, and Y. Bengio. 2014. Neural machine translation by jointly learning to align and translate. *ArXiv*, 1409.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin 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, Dubrovnik, Croatia. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, and 1 others. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*.

censusindia. 2011. District census handbook lawngtlai. Office of the Registrar General.

Nadir Durrani, Hassan Sajjad, Alexander Fraser, and Helmut Schmid. 2010. Hindi-to-Urdu machine translation through transliteration. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 465–474, Uppsala, Sweden. Association for Computational Linguistics.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy. PMLR.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Y. Bengio. 2015. On using monolingual corpora in neural machine translation.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Kazi Samin Mubasshir, Md Hasan, Madhusudan Basak, Mohammad Rahman, and Rifat Shahriyar. 2020. Not low-resource anymore: Aligner ensembling, batch filtering, and new datasets for bengali-english machine translation.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Tom Kocmi and Ondřej Bojar. 2018. Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine*

Translation: Research Papers, pages 244–252, Brussels, Belgium. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54. Association for Computational Linguistics. 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Langauge Technology (HLT-NAACL 2003) ; Conference date: 27-05-2003 Through 01-06-2003.

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.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Preprint*, arXiv:2001.08210.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Amena Mohsin. 2013. Language, identity and state. In Naeem Mohaiemen, editor, *Between Ashes and Hope: Chittagong Hill Tracts in the Blind Spot of Bangladesh Nationalism*, page 158. Drishtipat Writers' Collective.

Kanchon Kanti Podder, Ludmila Emdad Khan, Jyoti Chakma, Muhammad E.H. Chowdhury, Proma Dutta, Khan Md Anwarus Salam, Amith Khandakar, Mohamed Arselene Ayari, Bikash Kumar Bhawmick, S M Arafin Islam, and Serkan Kiranyaz. 2023. Selfchakmanet: A deep learning framework for indigenous language learning using handwritten characters. *Egyptian Informatics Journal*, 24(4):100413.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

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, Brussels, Belgium. Association for Computational Linguistics.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Mamdouh Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, Alexei Baevski, Yossi Adi, Xiaohui Zhang, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. 2023. Scaling speech technology to 1, 000+ languages. *ArXiv*, abs/2305.13516.

General Assembly resolution 44/25. 1989. *Convention on the Rights of the Child*, adopted and opened for signature, ratification and accession by general assembly resolution 44/25 of 20 november 1989 edition. United Nations.

Hammam Riza, Michael Purwoadi, Gunarso, Teduh Uliniansyah, Aw Ai Ti, Sharifah Mahani Aljunied, Luong Chi Mai, Vu Tat Thang, Nguyen Phuong Thai, Vichet Chea, Rapid Sun, Sethserey Sam, Sopheap Seng, Khin Mar Soe, Khin Thandar Nwet, Masao Utiyama, and Chenchen Ding. 2016. Introduction of the asian language treebank. In *2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA)*, pages 1–6.

Jonali Saikia and Mary Kim Haokip. 2023. Language endangerment with special reference to chakma. 3.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. *Preprint*, arXiv:1511.06709.

Statistics. 2023. Bangladesh bureau of statistics. 2021. "Table A-1.4 Ethnic Population by Group and Sex".

UnitedNation. *Convention on the Rights of Persons with Disabilities and Optional Protocol*. UN.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2005. Parallel corpora for medium density languages. *In Proceedings of the Recent Advances in Natural Language Processing (RANLP 2005)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. 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*, pages 483–498, Online. Association for Computational Linguistics.

Shiyue Zhang, Benjamin Frey, and Mohit Bansal. 2020. ChrEn: Cherokee-English machine translation for endangered language revitalization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 577–595, Online. Association for Computational Linguistics.

A Appendix

A.1 Additional Details on Expert-Based Data Collection

Prior to data collection, we conducted a pre-assessment to evaluate the feasibility of manual translation by volunteer Chakma speakers. The assessment revealed that participation and translation quality were highly sensitive to task complexity and time requirements. In particular, longer sentences significantly increased cognitive load and annotation time, frequently resulting in incomplete or inaccurate translations. These difficulties were exacerbated by the fact that many Bangla or English lexical items are either rarely used in Chakma or lack direct lexical equivalents, often requiring paraphrasing or explanatory reformulation. Based on these findings, we constrained source sentences to a length of 2 to 8 words to reduce annotation burden while maintaining sufficient linguistic content.

To mitigate these challenges and reduce annotator fatigue, we deliberately selected short sentences, with the probability of sentence selection peaking at 4-5 words and gradually decreasing toward both extremes. This distribution reflects an optimal trade-off between linguistic informativeness and annotation feasibility. Very short sentences (e.g., one word) were avoided due to limited contextual value, while longer sentences were excluded to minimize translation difficulty and error propagation.

The participants were predominantly young native Chakma speakers with functional bilingual proficiency in Bangla and English but limited formal training in translation. Despite their linguistic competence, many participants reported difficulty translating abstract or institutional terms, as Chakma remains primarily an oral language with limited standardized written usage. Furthermore, expert translators typically provided handwritten translations, which were later digitized by trained typists due to limited familiarity with Chakma script typing.

We discuss additional things in Appendix A.2 and A.5

A.2 Origins of Orthographic Variation in Chakma

Chakma lacks a widely accepted standardized grammar, resulting in substantial variation in spelling and syntactic structure across written sources. Historically, language experts and local shamans have documented the language using personal conventions without publishing formal grammatical guidelines. As a result, the same lexical items are often written using multiple valid spellings, leading to pervasive orthographic inconsistency. Disagreements among scholars have further hindered consensus on standard grammatical rules. These disagreements are also reflected in differences between Indian and Bangladeshi Chakma scholarly traditions. Such variability complicates data normalization and automatic evaluation in downstream NLP tasks.

A.3 Script-Bridging Transliteration: Mapping Coverage and Non-Direct Rules

This appendix provides the full mapping statistics and the handling rules for characters that do not admit an exact one-to-one correspondence between Chakma (CCP) and Bangla (BN). A summary of all non-direct substitutions in both directions is shown in Figure 3.

CCP→BN coverage. In the CCP→BN direction, all 10 Chakma digits have direct mappings. Among the core Chakma letters (vowels and consonants), 36 out of 37 characters map directly to Bangla equivalents. For diacritics, 18 out of 21 Chakma diacritical marks have direct mappings. The remaining Chakma characters correspond to prosodic or orthographic distinctions that do not have explicit representations in Bangla. Notably, one such character functions as a lengthening/extension marker: it modifies the pronunciation of an adjacent letter rather than introducing independent lexical content. Since Bangla lacks an equivalent graphemic mechanism for this feature, we normalize it during transliteration by preserving the base character without adding a distinct symbol in the Bangla rendering, thereby maintaining lexical meaning and the underlying pronunciation class. Moreover, the four Chakma characters without direct Bangla equivalents are extremely rare in contemporary Chakma usage and do not materially affect downstream translation quality.

BN→CCP coverage. In the reverse BN→CCP direction, all 10 Bangla digits have direct mappings. Out of 50 Bangla letters, 44 map directly to Chakma characters. Similarly, 10 out of 11 Bangla diacritics have corresponding Chakma equivalents. The remaining Bangla characters encode phonetic distinctions that are not contrastive in Chakma orthography and are therefore mapped to the closest Chakma counterparts that best preserve pronunciation and lexical identity.

Deterministic handling of non-direct characters. For characters without direct one-to-one correspondences in either direction, we apply deterministic substitutions based on closest phonetic similarity in the target script (Figure 3). Consequently, the transliteration system prioritizes content preservation over strict character reversibility: round-trip transliteration may introduce minor surface-level variation, but does not lead to semantic or lexical information loss (Section 7). Overall, the system remains predominantly a straightforward character mapping scheme, with a small number of rule-based phonetic normalizations applied only when exact script-level equivalence is unavailable.

A.4 Additional Experimental Details

Multiple Runs and Randomness To assess robustness to training stochasticity, we run all experiments three times with different random seeds (affecting initialization and minibatch order) and report mean and standard deviation. For in-context learning, we fix the demonstration set and repeat generation three times to quantify decoding variability. Therefore, reruns under the same configuration are expected to yield scores consistent with the reported mean ± standard deviation.

Normalization details On top of the normalization described by (Hasan et al., 2020), we apply a minimal and conservative normalization step uniformly to all text, including training data, model inputs, and output labels, across all experiments. For Chakma script, we merge a small number of rarely used, phonetically similar vowel variants into a common representation to reduce superficial spelling variation. This is analogous to collapsing long and short vowel variants in low-resource settings and is intended to simplify orthographic variation while preserving pronunciation and meaning. We also normalize all bracket symbols to parentheses in order to reduce sparsity in the data. These normalization steps are applied symmetrically and are not intended to alter semantic content or translation difficulty.

Additional details on metrics We report BLEU and chrF scores using `sacreBLEU` via the Hugging Face `evaluate` library with default settings: BLEU uses a maximum word n-gram order of 4 (BLEU-4), and chrF uses character n-grams of order 6.

SentencePiece Vocabulary Search We use SentencePiece (Kudo and Richardson, 2018) both for (i) vocabulary building and (ii) tokenization for SMT and NMT. As part of hyper-parameter optimization, we evaluate vocabulary sizes of 1,000, 2,000, 5,000, 10,000, and 20,000.

Training Hyper-Parameters and Optimization Settings We apply gradient clipping with max norm 1.0. We tune learning rates in {0.001, 0.005, 0.0001, 0.0005}, batch sizes in {8, 16, 32}, and training steps in {10,000, 15,000, 20,000}. Warmup steps are varied in {0, 2000, 4000}. We also tune label smoothing over {0.1, 0.2, 0.3, 0.4, 0.5}. The final tuned hyper-parameter configurations are reported in Table 7.

RNN with Attention: Architecture and Initialization For the RNN baseline, we use a public PyTorch seq2seq implementation.[12] The model incorporates Luong-style attention (Luong et al., 2015). We experiment with 1, 2, and 4 stacked recurrent layers, and consider hidden size and embedding size in {512, 1024}. Dropout is tuned in {0.1, 0.2, 0.3}. All RNN parameters are initialized from a normal distribution with mean 0 and standard deviation 0.1.

Transformer: Model Variants and Initialization For Transformer training, we follow the standard Transformer formulation (Vaswani et al., 2017). We explore MarianNMT-style Transformer[13] configurations available through HuggingFace implementations, and initialize weights with Glorot initialization (Glorot and Bengio, 2010). We vary the number of layers in {1, 2, 6}, attention heads in {1, 2, 6}, dropout in {0.1, 0.2, 0.3}, and feed-forward hidden dimensions in {512, 1024}.

Multilingual Formatting, and Oversampling In a multilingual training(MNMT), we prepend a target-language prefix tag to each input sentence:

[12] https://github.com/bentrevett/pytorch-seq2seq/tree/main

[13] https://huggingface.co/docs/transformers/model_doc/marian

<BN> for Bangla, <EN> for English, and <CCP> for Chakma. To mitigate imbalance, we oversample Chakma-involving parallel pairs to better balance all translation directions, following practices shown to improve multilingual performance (Johnson et al., 2017).

In-Context Learning Prompt Construction For in-context learning (ICL), demonstration examples are selected either uniformly at random or using character-level n-gram overlap to account for orthographic variation in Chakma. Character-level matching is used instead of word-level matching due to the absence of standardized spelling. The n-gram size was selected through limited manual experimentation using a small subset of the development data, due to computational budget constraints. This subset was used only for preliminary testing of retrieval behavior, and not for model selection or final evaluation. We evaluated a narrow range of values and fixed $n = 6$, which provided the most stable retrieval behavior in these tests.

mBART: Additional Details We do not apply iterative back-translation (IBT) or multilingual fine-tuning (MNMT) to mBART, as its plain fine-tuning performance is substantially lower than other models (Table 2, Section 6), making these extensions unlikely to provide meaningful improvements.

A.5 Interviews with Chakma Language Experts

We interviewed several scholars in Bangladesh to discuss the variants, for example, the number of characters, diacritics, rules, spelling patterns, etc. The scholars include Arjya Mitra, Injeb Chakma, Ananda Mohon Chakma, and Sugata Chakma. However, almost all of them suggested following the rules maintained by the members of the National Curriculum and Textbook Board of Bangladesh involved in writing the Chakma books for the pre-primary levels because their rules will be followed eventually. The most important rule from them that we followed in our transliteration codes from Bangla to Chakma, is that the core grapheme cannot have more than one diacritic attached to a consonant or a vowel. However, in India, this restriction is not maintained, rather more than one diacritic is seen frequently in their documents.

Title	Content	Samples
Ajanir dajan firana.docx	Story	206
Amader-Bari-2.pdf	Story	12
Amader-Bari-3.pdf	Story	23
Amader-gaye-dewar-pinon.pdf	Story	10
Amar-Charar-Boi.pdf	Poem	123
Amlokir-Gach.pdf	Story	27
Article 3rd Jamachug.docx	Story	194
Article 4th Furamon.docx	Story	194
Article 5th Pawr Murah.docx	Story	191
Bang-O-Puti-mach.pdf	Story	11
Banor-Berate-Eseche.pdf	Story	35
Banorer-Marfa-khaowa.pdf	Story	10
Bashir-soor.pdf	Story	9
Bie-Bari.pdf	Story	28
Bijhu.pdf	Story	28
Binoy Bikash Talukder20.docx	Poem	647
Binoy Dewan.docx	Poem	2004
Bizute-Berano.pdf	Story	12
Bone-Gie-Gach-Kata.pdf	Story	30
Boner-Mama.pdf	Story	11
Chader-Buri.pdf	Story	28
Chakma Dictionary app	Other	14928
Chakma Folktales app	Story	3765
Chakma Love song Uvagit.docx	Story	13
Chakma Text Book For Class-IV 2010 (IN Govt).docx	Textbook	1088
Chakma Text Book for Class-II 2010 (IN Govt).docx	Textbook	490
Chakma Text Book for Class-III 2010 (IN Govt).docx	Textbook	561
Chakma Text Book for Class-V 2010 (IN Govt).docx	Textbook	940
Chakma Text Book for Class-VI 2010 (IN Govt).docx	Textbook	1543
Chakma Text Book for Class-VII 2010 (IN Govt).docx	Textbook	1858
Chakma.docx	Article	136
Charar Boi-Chakma-Pages.pdf	Poem	31
Cijir Orago Boi-Chakma-Pages.pdf	Other	71
Cijir Talmiloni Kodatara-Chakma-Pages.pdf	Other	45
Cycle-e-Bazare-Jawa.pdf	Story	33
Dhanpudi.doc	Story	1278
Dudur-Kanna.pdf	Story	40
Dui-Bandhobir-Kotha.pdf	Story	16
Ghara Poja pire-Chakma-Pages.pdf	Other	4
H.F.Miller's Rangakura.docx	Story	90
Hotat-Agun.pdf	Story	12
Iskulo Akto-Chakma-Pages.pdf	Other	5
Jhimit-Ekhon-Bhalo.pdf	Story	42
Jhogra-Kora-Valo-Noi.pdf	Story	42
Kalo-and-Forshar-Kotha-1.pdf	Story	22
Kanamachi-Khela.pdf	Story	13
Karo-bipode-hasa-thik-na.pdf	Story	15
Kolar-Kotha-1.pdf	Story	11
Korgosher-sobji-bagan.pdf	Story	12
Lairang-er-nodi-par-howa.pdf	Story	13
Lao-er-Desh-Vromon.pdf	Story	44
Laz-kata-Banor.pdf	Story	12
Lobh-kora-valo-na.pdf	Story	16

Table 3: Names of the sources of our Chakma monolingual data with details (Part 1).

Title	Content	Samples
Mamar-Bari.pdf	Story	19
Mayer-Upadesh-1.pdf	Story	19
Meghla-Akash.pdf	Story	22
Mitar-Fuler-Bagan-1.pdf	Story	10
Moina-Pakhi-1.pdf	Story	16
Monar Sabon-Chakma-Pages.pdf	Story	36
Moni-Malar-Kotha-.pdf	Story	22
Monir-shopno-dekha.pdf	Story	14
Morog-Jhuti-Fool.pdf	Story	25
My Legha by Injeb Chakma.doc	Story	727
Nada-bhet-math for class I (IN Govt Tripura).docx	Textbook	878
Nanarakam-ghor.pdf	Story	14
Nirapod-pani-pan-korbo.pdf	Story	13
Ojhapador Chora-Chakma-Pages.pdf	Poem	30
Paka-Lichu.pdf	Story	19
Porichoy.pdf	Story	16
Projapoti-Ronger-Kotha.pdf	Story	12
Puti-Macher-Fal.pdf	Story	13
Rangdhanu.pdf	Story	20
Ranjuni for Class I (IN Govt) Tripura.docx	Textbook	1459
SRM 1st P. Bargang.docx	Poem	156
SRM 1st R. Krisnachura.docx	Poem	149
SRM 2nd P. Belwa Pawr.docx	Poem	259
SRM 2nd R. Chadarok.docx	Poem	76
Sanye-Pidhe-.pdf	Story	6
Shikkha Boi2017.docx	Poem	722
Shing-Macher-Kata.pdf	Story	36
Shiyal-er-Khang-Garang-Bazano.pdf	Story	19
Shrout.pdf	Story	8
Sial-mamar-school.pdf	Story	14
Sukorer-pat-batha-1.pdf	Story	12
Surjyer-Manush.pdf	Story	21
Tanybi.doc	Story	79
Tarum A Ranjuni-Chakma-Pages.pdf	Other	16
Teen-bondhur-golpo.pdf	Story	13
Text-Book-Chakma-pdf.pdf	Story	1405
Thurong-Barite-Raja.pdf	Story	43
Tin-bondhur-gacher-kotha.pdf	Story	15
Tiya-Pakhi-1.pdf	Story	23
chakma novel hlachinu.docx	Novel	1571
chedon akkan(10).pdf	Article	103
diarrhea-hole-ki-Korbo.pdf	Article	18
ghila khara class 3 p. 62.docx	Story	133
kajer-Kotha.pdf	Story	11
kochpanar rubo rega.docx	Story	151
mle- 2 ananda babu.docx	Poem	174
tin fagala-1.docx	Novel	1765
Changma Ekbacchya Kodha2.doc	Other	170
Chadi 2 Pojhodhe.docx	Novel	1209

Table 4: Names of the sources of our Chakma monolingual data with details (Part 2).

ASCII Font list of Chakma
BivunabaKhamaC
BijoygiriDPC
Udoy Giri
Alaam
Arjyaban
Chakma(SuJoyan)
Punong Jun

Table 5: Chakma ASCII fonts identified in our data sources and subsequently converted to the RibengUni (UTF-8) font as part of corpus normalization.

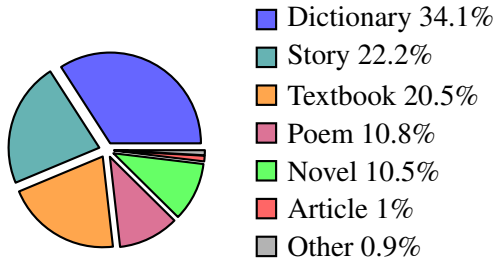

Figure 2: Distribution of Chakma monolingual data by content type. The corpus contains 42,783 monolingual samples collected from diverse sources, including dictionaries, stories, textbooks, poems, novels, and articles.

Class	Bangla	Direction	Chakma
Letter	শ	→	𑄯
Letter	ষ	→	𑄯
Letter	ড়	→	𑄰
Letter	ঢ়	→	𑄰
Letter	ঋ	→	𑄰
Letter	ৎ	→	𑄱
Diacritic	ৃ	→	𑄴
Letter	ওআ	←	𑄖
Diacritic	োআ	←	𑄘
Diacritic	োআ	←	𑄘
Diacritic	-	←	𑄙

Figure 3: Nearest-character substitutions used in the Chakma–Bangla transliteration system for characters without direct one-to-one mappings. These substitutions preserve semantic content and approximate pronunciation, while potentially neutralizing non-contrastive orthographic distinctions. The only entry marked with a dash (–) in the Bangla column corresponds to a rare Chakma prosodic lengthening marker that lacks an explicit Bangla graphemic equivalent and is normalized during transliteration. All four Chakma characters without direct Bangla counterparts are extremely rare in contemporary usage and have negligible impact on downstream translation quality.

You are given translation examples from Chakma to Bangla below:

Chakma Example 1: মরে নুয়া গরি পিজোর্ ন গরিবে
Bangla Example 1: আমাকে পুনরায় জিজ্ঞাসা করবে না ।

Chakma Example 2: গির্গিরানা
Bangla Example 2: কাঁপা

...
...

Chakma Example K: এ সভাগুনৎ কোরাম্ পুরেবাৎয়াই সরিক্ রাস্ট্রআনির্ তিনভাগর্ দিভাগ্ হাজির্ থা -পরিব ।
Bangla Example K: অংশগ্রহণকারী রাষ্ট্রের দুই -তৃতীয়াংশ উপস্থিতি ফোরাম হিসাবে বিবেচিত হবে ।

Provide the Bangla Translation of the Chakma provided below. Only provide the translation and do not output anything else.

Chakma Test 1: বলানয়ায়ান্ গুঝিবৈলো সুনানু মন্ত্রি কোহেয়দে , ধয় কত্তেদি সুনজুক্কানেদয়ই তে কাতার্ বেরা জেব ।
Chakma Test 2: সুনানু মন্ত্রা সেখ্ খাসিনা জধা ভিচ্চে আরব্ আমিরাদৎ (ইউএই) তিন্ দিনর্ সরাকারি পর্ভাচ্ বিদি এচ্চে রেদোৎ দেঝৎ লুহ্গেগি ।
...
...
Chakma Test 20: এ আলহটয়্ সুক্রবার্ পুলিঝর্ বেগ দাঙর্ চোক্ দিয়েবো (আইজিপি) জাবেদ্ পাটোয়ারি কোয়াহেদ , ভুক্ক্ গরিয়ে কিঐো জনি তুল্বিচ্ ন চাহ্না , সালেন্ পুলিসসুনে নিজে গিরোহচ্ ঠৈহোনে তুল্বিচ্ চাহ্নাক্ ।

Figure 4: Prompt format used for our few-shot in-context learning (ICL) experiments, illustrating the structure of source-target examples, task instructions, and test-time input.

System	BN→CCP				CCP→BN			
	Dev		Test		Dev		Test	
	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
BanglaT5	**10.9** ± 0.06	**36.3** ± 0.22	**02.7** ± 0.05	**30.6** ± 0.21	**24.4** ± 0.11	**44.6** ± 0.08	**11.7** ± 0.19	**37.9** ± 0.09
BanglaT5†	00.0 ± 0.02	00.5 ± 0.04	00.0 ± 0.00	00.5 ± 0.18	02.3 ± 0.04	14.5 ± 0.09	00.1 ± 0.02	14.3 ± 0.37
mT5-small	**08.5** ± 0.11	**32.2** ± 0.20	**02.0** ± 0.04	**25.8** ± 0.12	**14.1** ± 0.33	**33.9** ± 0.35	**04.6** ± 0.07	**27.2** ± 0.25
mT5-small†	00.2 ± 0.17	09.1 ± 0.91	00.0 ± 0.00	07.7 ± 0.23	00.8 ± 0.04	11.9 ± 0.99	00.1 ± 0.00	11.7 ± 0.37
GPT-4.1(R)	-	-	**01.6** ± 0.13	**30.4** ± 0.12	-	-	**16.5** ± 0.12	**48.2** ± 0.35
GPT-4.1(R)†	-	-	00.4 ± 0.12	18.3 ± 0.12	-	-	00.3 ± 0.10	18.2 ± 0.21

Table 6: Ablation study comparing models evaluated *with* and *without* transliteration on BN→CCP and CCP→BN translation. Rows marked with † indicate evaluation **without transliteration** (native Chakma script as input and output). Comparison includes BanglaT5 and mT5-small fine-tuned models, as well as GPT-based in-context learning (ICL) models with random sampling of 400 examples. Metrics report mean ± std for BLEU and chrF on the development and test sets. Removing transliteration results in near-zero performance across both fine-tuning and ICL, highlighting its necessity in this setting.

	Src. 1	জাতিসংঘের সংস্থাগুলো এবং প্রযুক্তি বিশেষজ্ঞদের কাছ থেকে জরুরি প্রস্তুতি
	Ref. 1	[illegible] (United Nations agencies and technical experts on emergency preparedness)
	Src. 2	জাতিসংঘের সংস্থাগুলো এবং প্রযুক্তি বিশেষজ্ঞদের কাছ থেকে জরুরি প্রস্তুতি , বসবাসযোগ্যতা এবং দ্বীপের সুরক্ষার বিষয়ে কোনো ছাড়পত্রের জন্য অপেক্ষা না করে রোহিঙ্গাদের সেখানে স্থানান্তরিত করায় সরকার তার বারবার করা প্রতিশ্রুতি ভঙ্গ করেছে ।
	Ref. 2	[illegible] (The government has also reneged on repeated promises to await clearance from United Nations agencies and technical experts on emergency preparedness, habitability, and safety of the island before relocating Rohingya there.)
SMT	Pred. 1	[illegible] (United Nations ...) [Correctly generates a named entity, but the output is not semantically interpretable.]
	Pred. 2	[illegible] (United Nations' General Secretary ...) [Correctly identifies a named entity, but fails to preserve sentence-level semantics.]
RNN	Pred. 1	[illegible] (United Nations' children and others) [Literal lexical translation is produced, but the output is not semantically interpretable.]
	Pred. 2	[illegible] (United Nations' all and others and why) [Literal lexical translation is produced, but the output is not semantically interpretable.]
Trans -former	Pred. 1	[illegible] (United Nations' United Nations' United Nations' United Nations' help help) [Degenerate repetitive generation with severe semantic degradation.]
	Pred. 2	[illegible] (United Nations' United Nations' all country country country country autistic person's rights rights rights and and and their their their own own own country country country) [Degenerate repetitive generation with severe semantic degradation.]
BanglaT5	Pred. 1	[illegible] (Preparedness required from United Nations agency and technical expert persons) [Semantically faithful with minor orthographic variation.]
	Pred. 2	[illegible] (The government has reneged on repeated promises to await clearance from United Nations agencies and technical experts on emergency preparedness, habitability, and safety of the island relocating Rohingya there.) [Semantically faithful to the reference with minor orthographic variation.]
GPT 4.1 (ICL)	Pred. 1	[illegible] (United Nations agencies and technical experts on emergency preparedness) [Semantically faithful to the reference, with lexical choices that differ slightly from BanglaT5 while preserving meaning.]
	Pred. 2	[illegible] (The government has reneged on repeated promises to await clearance from United Nations agencies and technical experts on emergency preparedness, habitability, and safety of the island before relocating Rohingya there.) [Semantically faithful to the reference, with lexical choices that differ slightly from BanglaT5 while preserving meaning.]

Figure 5: Qualitative examples of BN→CCP translations generated by different models on short and long sentences, illustrating differences in semantic adequacy and robustness across approaches. Parentheses () provide a literal English gloss of each model output for readability, while square brackets [] give a brief qualitative analysis comparing translation quality. Overall, BanglaT5 and GPT models using in-context learning (ICL) produce more semantically faithful translations than earlier SMT, RNN, and baseline Transformer approaches, particularly for longer sentences.

Src.	ইফতারের আগে বিশেষ মোনাজাতে দেশ ও জাতির অব্যাহত শান্তি, অগ্রগতি এবং সমৃদ্ধি কামনা করা হয়। (Before the iftar, a special munajat was offered seeking continued peace, progress and prosperity of the nation.)
T1.	𑄌𑄨𑄝𑄢𑄴𑄛𑄳 𑄢𑄩𑄃 𑄝𑄬𑄟𑄚𑄨 𑄝𑄧𑄢𑄨-𑄢𑄨 𑄇𑄚𑄨 𑄢𑄇 𑄝𑄇𑄛𑄳 𑄐𑄴 𑄘𑄧𑄚𑄃𑄃 𑄚𑄨, 𑄢𑄬𑄝𑄧𑄇 𑄢𑄇 𑄘𑄬𑄃𑄇 𑄌𑄬𑄛𑄧 𑄝𑄢𑄬𑄚𑄨 𑄝𑄢𑄳 𑄝𑄬𑄚𑄃𑄃।
T2.	𑄌𑄨𑄝𑄢𑄴𑄛 𑄢𑄩𑄃𑄇 𑄝𑄬𑄟𑄚𑄨 𑄇𑄚𑄨 𑄢𑄬 𑄝𑄇𑄛𑄳 𑄌𑄬𑄛𑄧𑄚𑄬𑄚𑄨 𑄚𑄬𑄛𑄧𑄚𑄨 𑄢𑄬𑄢𑄬𑄚𑄨 𑄢 𑄘𑄬𑄃𑄇 𑄢𑄬𑄚𑄬𑄚𑄬 𑄢𑄬𑄚𑄬𑄚𑄬 𑄝𑄢𑄳 𑄝𑄬𑄚𑄃𑄃।
T3.	𑄌𑄨𑄝𑄢𑄴𑄛𑄳 𑄢𑄬𑄝𑄬𑄚𑄨 𑄝𑄬𑄟𑄚𑄨 𑄝𑄬𑄟𑄚𑄨 𑄇𑄚𑄨 𑄢𑄬 𑄝𑄇𑄛𑄳 𑄝𑄬𑄚𑄬𑄚𑄬𑄚𑄨 𑄝𑄬𑄚𑄬𑄚𑄬 𑄝𑄢𑄳 𑄝𑄬𑄚𑄃𑄃।

Figure 6: Illustration of orthographic variation in Chakma, where multiple valid spellings of the same word appear across translations produced independently by three language experts from the same Bangla source. Identical colors highlight variant spellings of the same lexical item.

Parameter	RNN	Trans.	T5
Max Epochs	-	-	5
Max Train Steps	20000	20000	-
Warmup Steps/Ratio	4000	4000	0.1
Learning Rate	0.0005	0.0001	0.0005
Batch Size	16	32	16
Max Length	128	128	128
Optimizer	adam	adam	adam
Vocab size	2000	10000	-
Beam width	5	5	5
Clip gradient	1.0	1.0	-
Label Smoothing	0.2	0.5	0.3
d_model	-	512	-
dropout	-	0.2	-
layer_dropout	-	0.1	-
att_heads	-	1	-
ffn_dim	-	512	-
blocks	-	6	-
rnn_dropout	0.3	-	-
layer_normalization	True	-	-
layers	1	6	-
word_embedding	512	-	-
hidden_embedding	1024	-	-
weight_decay	-	-	0.01

Table 7: Final training hyperparameters selected based on validation performance for from-scratch RNN and Transformer models, and for fine-tuning pretrained T5-based models (BanglaT5 and mT5-small). mBART was fine-tuned using the same hyperparameter settings as T5.

System	#Ex.	Random Sampling				N-gram Similarity Sampling			
		BN→CCP		CCP→BN		BN→CCP		CCP→BN	
		BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
GPT-4.1	100	01.3 ± 0.13	28.7 ± 0.40	16.2 ± 0.23	47.8 ± 0.12	01.4 ± 0.17	30.4 ± 0.30	16.9 ± 0.31	48.7 ± 0.28
	200	01.3 ± 0.07	29.5 ± 0.20	**16.8** ± 0.53	47.8 ± 0.14	01.4 ± 0.15	30.5 ± 0.35	17.5 ± 0.28	49.1 ± 0.34
	400	**01.6** ± 0.13	**30.4** ± 0.12	16.5 ± 0.12	**48.2** ± 0.35	**01.5** ± 0.10	**31.3** ± 0.53	**17.8** ± 0.12	**49.6** ± 0.12
GPT-4.1-mini	100	00.9 ± 0.05	27.6 ± 0.26	10.3 ± 0.37	40.2 ± 0.27	**01.2** ± 0.10	28.6 ± 0.35	**11.1** ± 0.50	**40.7** ± 0.30
	200	**01.1** ± 0.14	28.1 ± 0.31	**10.9** ± 0.20	**40.6** ± 0.12	01.1 ± 0.02	28.5 ± 0.24	**11.1** ± 0.03	**40.7** ± 0.10
	400	**01.1** ± 0.06	**28.4** ± 0.19	**10.9** ± 0.19	40.0 ± 0.58	**01.2** ± 0.01	**28.7** ± 0.25	10.5 ± 0.54	40.4 ± 0.47
GPT-o4-mini	100	01.3 ± 0.04	28.6 ± 0.29	**12.8** ± 0.06	42.7 ± 0.03	01.1 ± 0.05	29.2 ± 0.14	**12.6** ± 0.42	**42.8** ± 0.29
	200	01.5 ± 0.02	28.3 ± 0.51	**12.8** ± 0.21	**42.9** ± 0.16	01.4 ± 0.04	**29.8** ± 0.18	12.4 ± 0.43	42.2 ± 0.60
	400	01.5 ± 0.03	**29.7** ± 0.03	12.5 ± 0.47	**42.9** ± 0.08	01.2 ± 0.07	29.7 ± 0.14	12.2 ± 0.27	42.7 ± 0.48

Table 8: In-context learning (ICL) performance of different GPT variants under identical experimental settings. Results compare random and n-gram similarity-based sampling of in-context examples across varying numbers of demonstrations (#Ex.). BLEU and chrF are reported as mean ± standard deviation for BN→CCP and CCP→BN translation. The table enables a controlled comparison of LLM variants under the same ICL conditions. Overall, ICL performance improves as the number of in-context examples increases, with gains becoming more consistent at 200-400 demonstrations across models, and n-gram similarity-based sampling generally yielding stronger results than random selection.

System	EN→CCP		CCP→EN		BN→CCP		CCP→BN	
	BLEU	**chrF**	**BLEU**	**chrF**	**BLEU**	**chrF**	**BLEU**	**chrF**
BanglaT5	01.2 ± 0.06	23.0 ± 0.28	06.5 ± 0.36	28.5 ± 0.32	02.4 ± 0.11	29.2 ± 0.32	12.8 ± 0.12	38.5 ± 0.61
mT5-small	00.2 ± 0.01	11.1 ± 0.88	01.0 ± 0.18	15.6 ± 0.34	01.5 ± 0.05	24.1 ± 0.73	03.0 ± 0.10	23.5 ± 0.34

Table 9: Test-set performance of multilingual fine-tuned models on EN↔CCP translation using BanglaT5 and mT5-small. BLEU and chrF are reported as mean ± standard deviation. Results for BN↔CCP are shown for reference to contrast multilingual performance with the bilingual setting.

Round	BN→CCP→BN		CCP→BN→CCP	
	BLEU	**chrF**	**BLEU**	**chrF**
1	41.55	79.32	38.37	79.69
2	99.97	99.99	97.61	99.46
3	100.00	100.00	100.00	100.00

Table 10: Round-trip transliteration quality up to the third iteration on the Benchmark set. Scores are reported as BLEU and chrF on benchmark sentences and show convergence after two rounds.

System	BN→CCP		CCP→BN	
	BLEU	chrF	BLEU	chrF
GPT-4.1	00.5 ± 0.09	21.1 ± 1.44	15.5 ± 0.49	45.9 ± 0.45
GPT-4.1-mini	00.6 ± 0.06	24.2 ± 0.71	09.2 ± 0.13	38.9 ± 0.02
GPT-o4-mini	00.6 ± 0.03	22.6 ± 1.41	12.2 ± 0.26	42.2 ± 0.23

Table 11: Zero-shot in-context learning ablation showing translation performance with no in-context examples for GPT-4.1, GPT-4.1-mini, and GPT-o4-mini on BN→CCP and CCP→BN. Results are reported as mean ± standard deviation for BLEU and chrF.