JU-NLP: Improving Low-Resource Indic Translation System with Efficient LoRA-Based Adaptation

Abstract

Low-resource Indic languages such as Assamese, Manipuri, Mizo, and Bodo face persistent challenges in NMT due to limited parallel data, diverse scripts, and complex morphology. We address these issues in the WMT 2025 shared task by introducing a unified multilingual NMT framework that combines rigorous language-specific preprocessing with parameter-efficient adaptation of large-scale models. Our pipeline integrates the NLLB-200 and IndicTrans2 architectures, fine-tuned using LoRA and DoRA, reducing trainable parameters by over 90% without degrading translation quality. A comprehensive preprocessing suite including Unicode normalization, semantic filtering, transliteration, and noise reduction ensures high-quality inputs, while script-aware post-processing mitigates evaluation bias from orthographic mismatches. Experiments across English↔Indic directions demonstrate that NLLB-200 achieves superior results for Assamese, Manipuri, and Mizo, whereas IndicTrans2 excels in English↔Bodo. Evaluated using BLEU, chrF, METEOR, ROUGE-L, and TER, our approach yields consistent improvements over baselines, underscoring the effectiveness of combining efficient fine-tuning with linguistically informed preprocessing for low-resource Indic MT.

1 Introduction

Low-resource Indic languages such as Assamese, Manipuri, Mizo, and Bodo pose significant challenges for **N**eural **M**achine **T**ranslation (NMT) due to data scarcity, script diversity, and linguistic complexity, often leading to suboptimal performance (Kunchukuttan, 2020a; Ramesh et al., 2023; Team et al., 2022a). This work aims to address these limitations by developing an efficient, parameter-optimized fine-tuning framework tailored for such underrepresented languages in the WMT 2025 shared task.

To address these gaps, we introduce a unified multilingual NMT pipeline tailored for low-resource Indic languages, combining robust preprocessing with parameter-efficient fine-tuning methods. We integrate **N**o **L**anguage **L**eft **B**ehind (NLLB-200) model (Team et al., 2022a) and **IndicTrans**2 (Ramesh et al., 2023) model, fine-tuning them using **L**o**w**-**R**ank **A**daptation (LoRA) as proposed by Hu et al. (2021a) and Weight-**D**ecomposed **L**o**w**-**R**ank **A**daptation (DoRA) as discussed by Zhao et al. (2023) to optimize performance while maintaining computational efficiency. Our preprocessing pipeline includes Unicode normalization, semantic filtering, transliteration (Kunchukuttan, 2020a), and noise reduction, ensuring high-quality input data for training. NLLB-200, with its extensive multilingual coverage, is adapted for English↔Assamese, Manipuri, and Mizo, while IndicTrans2, designed specifically for Indic languages, is fine-tuned for English↔Bodo to leverage its architectural strengths in low-data settings. The methodology ensures fair model comparison by maintaining consistent hyperparameters and evaluation settings across all language pairs, with key contributions lying in the combination of efficient fine-tuning, language-specific preprocessing, and script normalization for Indic NMT.

Our contributions include: (1) the first systematic application of LoRA/DoRA to NLLB-200 and IndicTrans2 for low-resource Indic languages, reducing trainable parameters by over 90% without sacrificing translation quality; (2) a novel preprocessing framework addressing script diversity and data noise, critical for morphologically complex languages; and (3) a comprehensive evaluation using BLEU (Papineni et al., 2002a), chrF (Popović, 2015), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004), and TER (Snover et al., 2006) metrics, demonstrating significant improvements over baseline approaches.

2 Related Work

Early work on translation involving Indic languages predominantly used statistical methods and ad-hoc bilingual corpora. For example, Koehn (2005a) introduced the *Europarl* corpus for SMT, but no comparable large-scale corpus existed for Indian languages. In practice, government and academic groups built phrase-based systems on much smaller data. India's *TDIL* mission developed the *Sampark* and *Anuvadaksha* translation programs by training phrase-based SMT models on limited domain-specific corpora. Similarly, Kunchukuttan and Bhattacharyya (2014) compiled *Sata Anuvadak*, a set of 110 SMT systems across Indian language pairs. These efforts established early benchmarks but exposed severe limitations due to data scarcity and domain mismatch.

With the advent of neural models, encoder–decoder architectures with attention (Bahdanau et al., 2015) and Transformers (Vaswani et al., 2017) became standard. Researchers trained RNN and then Transformer-based NMT systems for English–Hindi and other Indic pairs, often using byte-pair encoding and shared vocabularies. Multilingual and zero-shot strategies (Johnson et al., 2017) enabled parameter sharing across related languages, benefiting extremely low-resource pairs. Shared multilingual models improved translation quality through inductive transfer, as shown in early WMT shared tasks. Indic-to-Indic multilingual training further enhanced performance in cases of limited parallel data.

In recent years, large multilingual pre-trained models have been employed for Indic MT. Models like mBART (Liu et al., 2020) and mT5 (Xue et al., 2021) provide off-the-shelf improvements, even for Indian languages. In parallel, Indic-specific models such as IndicBART (Dabre et al., 2022) and IndicTrans2 (Ramesh et al., 2023) have emerged. These models were trained on carefully normalized Indic corpora and have shown superior performance in low-resource translation. IndicTrans2, in particular, supports translation across all 22 scheduled Indian languages and 462 Indic language pairs, making it one of the most comprehensive Indic MT systems.

More recently, ultra-large multilingual models and efficient fine-tuning methods have influenced this domain. The NLLB-200 model (Team et al., 2022b) introduced a massively multilingual architecture covering 200 languages, with strong performance on low-resource Indic pairs. To adapt such models efficiently, LoRA (Hu et al., 2021b) and DoRA (Zhao et al., 2023) have been proposed, drastically reducing fine-tuning cost while preserving performance. Finally, preprocessing methods such as Unicode normalization, script unification, and transliteration (Kunchukuttan, 2020b) have been shown to significantly enhance translation quality for Indic languages. These developments form the foundation for recent SOTA systems tailored to low-resource Indic MT.

3 Analysis of Dataset

For the machine translation experiments, we utilized the **WMT** 2025 corpus divided into two categories: **Category-1** (English $\leftrightarrow$ {Assamese, Mizo, Manipuri}) with moderate training data availability, and **Category-2** (English $\leftrightarrow$ Bodo) with limited training data. The following sections detail each language pair's parallel corpus specifications.

Table 1: Parallel sentences dataset statistics for both category -1 and 2.

Lang Pair	Script	Dataset	Parallel sents
En - As	Bengali	Training Validation Test	50000 2000 2000
En - Mni	Bengali	Training Validation Test	21687 1000 1000
En - Lus	Latin	Training Validation Test	50000 1500 2000
En - Bodo	Devanagari	Training Validation Test	13693 1000 1000

Table 1 summarizes the dataset sizes and scripts used for each language pair. The pairs En-As and En-Lus have the largest training sets ($50k$ sentences each), and the smallest ones are En-Mni and En-Bodo ($21,687$ and $13,693$ sentences, respectively). All language pairs are divided into validation and test sets, where En-As and En-Lus have a larger test set ($2,000$ sentences each), followed by En-Mni and En-Bodo ($1,000$ sentences each). The scripts are different by language, using Bengali for As and Mni, Latin for Lus, and Devanagari for Bodo.

Table 2 shows sentence-level statistics of the parallel corpora and illustrates the observed linguistic

Table 2: Sentence-level statistics for parallel corpora across four Indic language pairs.

Lang Pair	Avg. Sent. Length	Pearson Correlation	Unique Chars
En - As	En: 95.12 As: 91.29	0.7288	En: 137 As: 187
En - Mni	En: 102.79 Mni: 103.70	0.9447	En: 145 Mni: 177
En - Lus	En: 95.81 Lus: 97.73	0.8843	En: 119 Lus: 136
En - Bodo	En: 96.07 Bodo: 101.77	0.9377	En: 114 Bodo: 144

differences in the language pairs. The average number of words in an English sentence (En) ranges from 95.12 (En-As) to 102.79 (En-Mni). On the contrary, for target languages, the average number of words in a sentence is nearly the same or slightly longer with Manipuri (Mni) at 103.70 and Bodo at 101.77. The Pearson correlation coefficients, which measure the degree of alignment of sentence lengths of English with the target languages, show that En-Mni (0.9447) and En-Bodo (0.9377) have almost a perfect linear relationship, indicating highly consistent translation lengths. In contrast, En-As is least correlated (0.7288), indicating that more may vary concerning how sentence lengths are mapped in the languages. Additional script complexity of the unique character count is indicated by the number of characters for Assamese (As: 187), Manipuri (Mni: 177), Bodo (144), Mizo (Lus: 136). Together, these statistics emphasize the diversity of languages in the data, which may bear on the modeling of translators, especially for low-length-correlated languages or languages with a richer character set.

4 Methodology and Implementation Details

4.1 Data Preprocessing

- ***Unicode normalization*** is essential for machine translation in Indic languages because it ensures consistent text representation by converting multiple Unicode forms into a standardized format, improving tokenization, reducing noise, and enhancing alignment in parallel data. We have used `IndicNormalizer`[1] for Indic languages like Assamese and `unicodedata`[2] **N**ormalization **F**orm-**K** **C**anonical Composition (NFKC) normalizer for English language.

- ***Deduplication*** removes duplicate sentence pairs from parallel corpora, maximizing data utility for low-resource Indic machine translation. This is implemented by Python's built-in library `set()`, which removes duplicate sentence pairs from datasets.

- ***Ratio Filtering*** is essential in machine translation to ensure balanced sentence-length pairs by removing extreme mismatches, which could otherwise introduce noise and misalignment during training. Here, the implementation checks if the **word-count ratio** falls within `0.5` to `2.0`, retaining only pairs where the target sentence is neither half nor double the source length, thus preserving linguistically plausible alignments (Koehn, 2005b).

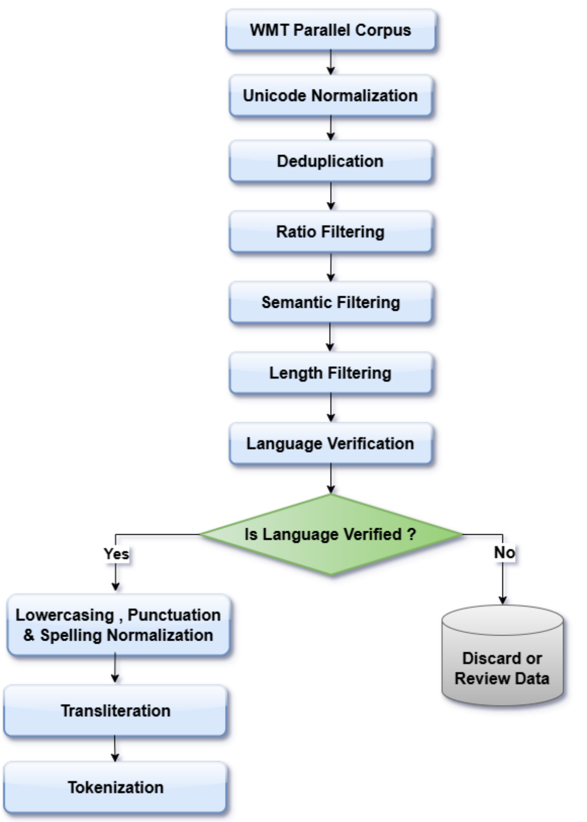

Figure 1: Workflow diagram of proposed data preprocessing pipeline.

[1] https://github.com/anoopkunchukuttan/indic_nlp_library

[2] https://docs.python.org/3/library/unicodedata.html

- ***Semantic filtering*** is crucial for Indic language machine translation to remove poorly aligned bilingual pairs that share surface-level similarities but differ in meaning. This is implemented using `LaBSE` (Feng et al., 2022) through cosine similarity scoring between sentence embeddings, where pairs scoring below a `0.75` threshold are excluded from training data to preserve semantic integrity.

- ***Length filtering*** is essential for machine translation to exclude excessively long sentences that may exceed model context limits or contain noisy data. This is implemented through a simple character count check (`150` words maximum per sentence) applied uniformly to both source and target texts.

- ***Language filtering***: To maintain high-quality, language-specific data for low-resource Indic machine translation, we employ FastText's pretrained language identification model (`ft_model`)(Joulin et al., 2017) to filter out noisy or mixed-language text. The sentences that are not confidently predicted as the target language are removed from the training corpus. Suspicious samples are retained for manual review to either: (1) salvage valuable translation pairs, or (2) analyze common noise patterns that could inform future data collection (Caswell et al., 2019).

- ***Text normalization***: We perform lowercasing, punctuation standardization, and spelling normalization (handling common orthographic variants) to reduce vocabulary sparsity. Aggressive noise removal eliminates HTML tags, non-linguistic symbols, and irregular whitespace, particularly crucial for noisy user-generated content in low-resource languages like Assamese.

- ***Transliteration*** is essential for handling named entities and rare words in low-resource Indic language machine translation. We implement a selective transliteration pipeline using `spaCy`[3] for tokenization and **N**amed **E**ntity **R**ecognition(NER), identifying words with frequency less than or equal to `2` or labeled as named entities. These words are transliterated from English to Indic scripts such as Assamese, Manipuri, and Mizo using the `IndicTransliteration` library[4], via the **H**arvard-**K**yoto (HK) scheme. This preserves phonetic structure and improves source-target alignment, enhancing overall translation quality.

- ***Tokenization*** splits text into subword units, crucial for handling morphologically rich Indic languages by addressing vocabulary sparsity and **O**ut-**o**f-**V**ocabulary (OOV) issues. For Assamese, Manipuri, and Mizo, we use Facebook's `NLLB-200-3.3`B tokenizer with a forced **B**eginning **O**f **S**equence (BOS) token for target language specification. For Bodo, we employ AI4Bharat's `Indictrans2` tokenizer, which supports multiple Indic languages via subword segmentation. Both tokenizers ensure compatibility with their respective Seq2Seq models by setting padding tokens dynamically.

4.2 Approach

This work utilizes the **WMT dataset** provided by the organizers. Consistent with established methodology for low-resource NMT, the data underwent preprocessing (detailed in Section 3) before model input to optimize translation quality for the target Indic languages. Given the focus on low-resource languages, specifically **Assamese, Manipuri, Mizo, and Bodo**, the model training pipeline is designed to leverage existing multilingual capabilities. In this study, two **s**tate-**o**f-**t**he-**a**rt (SOTA) open-source multilingual NMT models with pre-trained Indic language support are evaluated. Both models are subsequently fine-tuned on the preprocessed WMT dataset using LORA for parameter efficiency. Model selection is determined by comparative evaluation across standard automatic metrics: *BLEU*, *chrF*, *METEOR*, *ROUGE-L*, and *TER*.

The NLLB-200 model, developed by Meta AI, is a 3.3 billion-parameter multilingual sequence-to-sequence transformer that supports translation across 200 languages, including many low-resource ones, achieving SOTA performance. To fine-tune this model efficiently while preserving its generalization capabilities, we employ **P**arameter-**E**fficient **F**ine-**T**uning (PEFT) as discussed by Xu et al. (2023) via LoRA. This approach avoids full-model fine-tuning by instead injecting trainable

[3] `https://github.com/explosion/spaCy`

[4] `https://github.com/indic-transliteration/indic_transliteration`

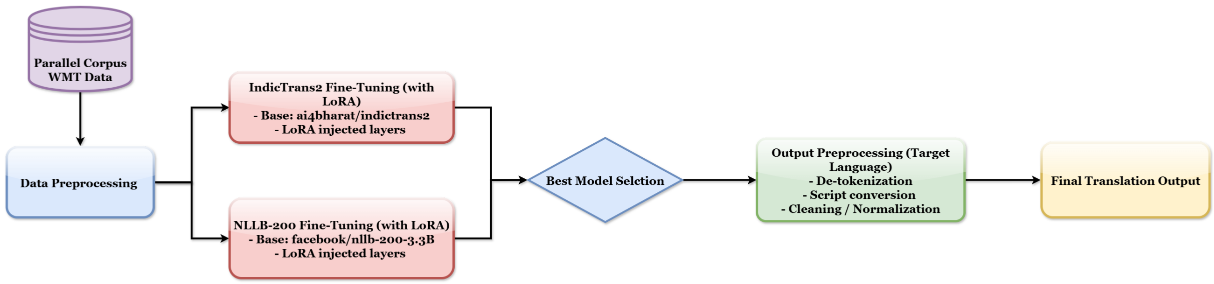

Figure 2: Bird's Eye View of the Proposed Approach

low-rank matrices into the transformer's attention layers, drastically reducing the number of trainable parameters while maintaining strong downstream task performance. The LoRA configuration is applied to the query, key, value, and output projection layers (`q_proj`, `k_proj`, `v_proj`, `o_proj`) of the NLLB-200 model. We set the rank (r) of the low-rank matrices to 64, with a scaling factor `lora_alpha` (α) of 128 to balance adaptation strength. A dropout rate of 0.1 is applied to the LoRA layers for regularization, and no additional bias terms are introduced. The model is then converted into a PEFT model, and all trainable parameters are logged before transferring the model to a CUDA-enabled 2$\times$ T4 Tesla GPU for accelerated training.

To handle variable-length sequences efficiently, we use a data collator specifically designed for sequence-to-sequence tasks. This collator dynamically pads input sequences to the longest length in each batch while ensuring padding aligns to multiples of 8 for optimal hardware utilization (Wolf et al., 2020). Label padding tokens (set to -100) are masked to exclude them from loss computation during training (Lewis et al., 2020). The training process leverages mixed-precision (FP16) arithmetic via the `Seq2SeqTrainer` from the Hugging Face Transformers library (Wolf et al., 2020). We employ a global batch size of 8, achieved through a per-device batch size of 4 and 2 gradient accumulation steps, balancing training stability (Micikevicius et al., 2018).

The optimization process uses AdamW with fused CUDA kernels (`adamw_torch_fused`), configured with momentum parameters $\beta_1 = 0.9$ and $\beta_2 = 0.98$ (Loshchilov and Hutter, 2019). The learning rate follows a cosine decay schedule, starting from 3×10^{-5} with 1000 warmup steps to ensure stable early training (Loshchilov and Hutter, 2016). Model checkpoints are saved at the end of each epoch, with the best model selected based on BLEU score (higher is better) (Papineni et al., 2002b). To improve evaluation efficiency, the trainer is configured to generate predictions during validation, enabling direct computation of translation metrics. To optimize memory efficiency, we disable caching (`model.config.use_cache = False`), enabling gradient checkpointing at the cost of modest recomputation (Chen et al., 2016). The complete training system integrates our LoRA-adapted NLLB-200 model with dynamic batching and automated evaluation, maintaining multilingual capabilities while specializing for target domains. This approach enables efficient adaptation of the 3.3B-parameter model, particularly valuable for low-resource languages where data efficiency is critical (Team et al., 2022a). The implementation demonstrates practical fine-tuning of massive multilingual models within resource constraints, balancing computational feasibility with translation quality.

On the other hand, the IndicTrans2, another state-of-the-art multilingual NMT model developed by AI4Bharat, supports translation between English and all 22 Indian languages, as well as direct Indic-to-Indic translation across 462 language pairs. It is optimized for high accuracy, long-context translation with both large (1.1B) and distilled (211M) model variants. It is fine-tuned using the same PEFT-LoRA methodology applied to NLLB-200. Identical LoRA hyperparameters (rank $r = 64$, $\alpha = 128$) target the query/key/value projections and dense layers, with DORA enhancing adaptation stability. We retain the 8-bit quantization strategy and FP16 mixed-precision training, but reduce gradient accumulation steps to 2 (effective batch size 8) due to the model's smaller footprint. The cosine learning rate schedule (3×10^{-5} peak, 500 warmup steps) and AdamW fused optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.98$) mirror the NLLB configuration, as does the BLEU-optimized checkpointing regime. Dynamic

Table 3: Evaluation metrics (BLEU, METEOR, ROUGE-L, chrF, and TER) for translation directions from English to four low-resource Indic languages for the evaluation dataset.

Language Pair	BLEU	METEOR	ROUGE-L	chrF	TER
en-as	17.5352	0.4223	0.0073	57.7459	71.1716
en-mni	4.1514	0.1554	0.0113	43.8669	93.1607
en-lus	15.8280	0.4193	0.5480	51.9998	69.0074
en-bodo	19.7083	0.4549	0.1694	62.4723	64.9709

Table 4: Evaluation scores (BLEU, METEOR, ROUGE-L, chrF, TER, and Cosine Similarity) for Indic-to-English translation directions for evaluation dataset.

Language Pair	BLUE	METEOR	ROUGE-L	chrF	TER	Cosine Similarity
As-En	0.3715	0.0127	0.0224	14.2593	116.7097	0.0388
Mni-En	8.1004	0.4798	0.4947	49.5997	100.2915	0.7974
Lus-En	12.2975	0.5778	0.6198	58.1381	78.8102	0.8888

batching via `DataCollatorForSeq2Seq` maintains padding efficiency, while disabled caching ensures memory headroom on T4 GPUs. This consistent approach allows fair comparison between the two SOTA multilingual systems while respecting their architectural differences.

In our evaluation pipeline, we adopt a systematic approach to compute evaluation metrics for assessing the translation quality of the two models. Before evaluation, the model-generated text is preprocessed once more to enhance the reliability of metric computation for the target language. After obtaining predictions and corresponding reference labels, both sequences are decoded using the tokenizer, with special tokens skipped during decoding. To ensure compatibility with BLEU and other metrics and to correctly handle padding label tokens, marked as -100 are replaced with the tokenizer's padding token ID. A key component of our implementation is the use of the `indic_transliteration` library (Kunchukuttan, 2020b), which converts the predicted text into the appropriate target language script. This transliteration step is crucial because, in the case of the **IndicTrans**2 model, the outputs are internally generated in the Devanagari script. In contrast, the reference translations are provided in native Indic scripts. Without this conversion, evaluation metrics would be skewed due to script mismatches rather than actual translation errors. Following transliteration, the decoded sequences are post-processed by removing extraneous whitespace, and evaluation is carried out using HuggingFace's `evaluate` toolkit (Lhoest et al., 2021), which provides robust and script-aware translation metrics for Indic languages.

5 Results and Discussion

We evaluate the translation quality of the fine-tuned models using a suite of established automatic evaluation metrics, with results presented in Tables 3 and 4. These results offer key insights into the relative difficulty and success of translating between English and four underrepresented Indic languages (i.e., Assamese, Manipuri, Mizo, and Bodo) in both directions.

Table 3 reports the evaluation results for English-to-Indic translation across four low-resource languages: Assamese, Manipuri, Mizo, and Bodo. Among these, the English-to-Bodo direction achieves the highest scores across multiple metrics, BLEU (19.70), METEOR (0.4549), and chrF (62.47), indicating superior translation adequacy and fluency under the proposed approach. For final output generation, model selection was based on a comparative analysis of evaluation scores obtained from IndicTrans2 and NLLB-200. The results show that NLLB-200 consistently outperforms IndicTrans2 for English-to-Assamese, Manipuri, and Mizo translations, whereas for the English-to-Bodo direction, IndicTrans2 demonstrates a clear advantage, yielding better translation quality.

Table 4 presents the evaluation metrics for translations from Indic languages to English. Among the language pairs, the Lus-En direction exhibits the strongest performance across nearly all metrics, BLEU (12.29), METEOR (0.5778), ROUGE-L (0.6198), chrF (58.13), and cosine similarity (0.8888), indicating high lexical and semantic alignment. In this translation direction, it was observed that the NLLB-200 model consistently outperforms IndicTrans2 for all three languages: Assamese, Manipuri, and Mizo.

Table 5: Relative performance ranks of translation directions based on evaluation metrics (lower rank indicates better performance).

Lang Pair	Rank
en-as	3
as-en	10
en-mni	2
mni-en	1
en-lus	1
lus-en	1
en-bodo	3

The relative performance ranks of translation directions based on evaluation metrics are presented in Table 5, where a lower rank indicates better performance and higher quality translation.

6 Conclusion

This study presents a comprehensive investigation into improving machine translation quality for low-resource Indic languages through parameter-efficient fine-tuning of large multilingual models. Leveraging LoRA and DoRA techniques, we fine-tuned both the NLLB-200 and IndicTrans2 models on a curated and rigorously filtered WMT2025 dataset. Our extensive preprocessing pipeline, tailored to address the idiosyncrasies of Indic languages, proved essential in ensuring clean and semantically aligned parallel corpora. The empirical results underscore that while NLLB-200 exhibits superior performance across most language pairs and metrics, especially in English-to-Indic and Indic-to-English directions involving Assamese, Manipuri, and Mizo, IndicTrans2 offers competitive results and even outperforms NLLB-200 in the English-to-Bodo direction.

Notably, our integration of script-aware post-processing and selective transliteration was instrumental in achieving faithful metric evaluations, avoiding script mismatch penalties that would otherwise misrepresent model performance. These findings not only validate the efficacy of LoRA-based adaptation in low-resource settings but also highlight the value of task-specific linguistic preprocessing for Indic languages. Our comparative benchmarking, involving multiple metrics, reveals the nuanced translation difficulty across language pairs and emphasizes the importance of direction-aware evaluations in multilingual NMT research.

Limitations

The WMT 2025 corpora, while suitable for benchmarking, are inherently limited in scale and domain diversity for certain language pairs, particularly English–Bodo and English–Manipuri. This scarcity restricts the models' ability to generalize to informal, noisy, or domain-specific contexts.

Although the preprocessing pipeline is comprehensive, fixed thresholds in semantic filtering and transliteration heuristics may inadvertently remove valid rare sentences or alter named entities. Subtle linguistic phenomena such as dialectal variation and code-mixing remain insufficiently addressed.

Methodologically, the study is restricted to LoRA and DoRA-based fine-tuning of NLLB-200 and IndicTrans2. Although this approach ensures parameter-efficient adaptation, it does not investigate other model architectures or combined training strategies that may more effectively address unique linguistic characteristics. Similarly, the exclusive use of automatic metrics provides reproducible benchmarks but offers limited insight into true semantic quality or culturally appropriate translations.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*.

Raj Dabre et al. 2022. Indicbart: A pre-trained model for indic languages. In *Proceedings of LREC*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021a. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Edward J. Hu et al. 2021b. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Melvin Johnson, Mike Schuster, Quoc V Le, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. In *Transactions of the Association for Computational Linguistics*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Philipp Koehn. 2005a. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, pages 79–86.

Philipp Koehn. 2005b. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Anoop Kunchukuttan. 2020a. The indic nlp library. Accessed: 2025-07-30.

Anoop Kunchukuttan. 2020b. The indic nlp library. Accessed: 2025-07-30.

Anoop Kunchukuttan and Pushpak Bhattacharyya. 2014. Sata anuvadak: Tackling multiway translation for indian languages. In *Proceedings of WAT*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Quentin Lhoest, Benjamin Minixhofer, Siddhartha Bandyopadhyay, et al. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu et al. 2020. Multilingual denoising pre-training for neural machine translation. In *Transactions of the Association for Computational Linguistics*.

Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR)*.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. 2018. Mixed precision training. In *International Conference on Learning Representations (ICLR)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002a. 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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002b. 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. ACL.

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.

Gowtham Ramesh, Vishrav Chaudhary, Divyanshu Kakwani, Sai Praneeth Golla, Abhishek Philip, et al. 2023. Indictrans2: Towards high-quality and efficient multilingual translation for indic languages. *arXiv preprint arXiv:2304.09105*.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Celebi, and et al. 2022a. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

NLLB Team et al. 2022b. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. 2017. Attention is all you need. In *NeurIPS*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, and Jamie Brew. 2020. Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu-Lee Wang. 2023. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *CoRR*, abs/2312.12148.

Linting Xue et al. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *NAACL*.

Lianmin Zhao, Shrimai Prabhumoye, Chen Shao, Yihong He, Xiaodong Ma, et al. 2023. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2306.11695*.