MaitH 1.0: A Parallel Corpus and Baseline for Low-Resource Maithili-Hindi Translation

Anonymous ACL submission

Abstract

Maithili is one of the 22 official languages recognized in the Indian Constitution. The literature of Maithili is rich; however, due to current socio-political changes, the language is on the verge of extinction. Therefore, it is crucial to develop a corpus for low-resource Indic languages like Maithili to ensure that the dream of "No Language Left Behind" (NLLB) is realized. With this in mind, we contribute a corpus (1,05,600 sentences) containing both manually curated and synthetically generated. Additionally, we propose a strong baseline on the Maithali-Hindi pair using our data, surpassing the baseline achievable through existing NLLB data.

1 Introduction

Machine translation (MT) has witnessed significant advancements over the past decade, driven largely by the availability of extensive parallel corpora and sophisticated models. However, these advancements are predominantly focused on high-resource languages, leaving many low-resource languages with limited or no effective translation systems. Maithili, a language spoken by over 22M people (according to Wiki) primarily in the eastern regions of India and the southern plains of Nepal, is one such low-resource language. Despite its rich linguistic heritage and substantial speaker base, Maithili remains underrepresented in the realm of natural language processing (NLP), particularly in machine translation.

The development of effective translation systems for low-resource languages like Maithili is crucial for several reasons. First, it helps in preserving linguistic diversity by enabling communication between speakers of different languages. Second, it provides access to information and services for speakers of these languages, contributing to social and economic inclusion. Finally, it adds to the global corpus of linguistic data, which is essential for studying and understanding human languages.

Several studies have focused on building translation systems for Indian languages, particularly those with limited resources. INDICNLP Project (Kunchukuttan, 2020) is a notable initiative to develop NLP resources and tools for Indian languages. It includes datasets, word embeddings, and other linguistic resources for multiple Indian languages, including low-resource languages. Researchers have created bilingual and multilingual corpora for Indian languages, which serve as essential resources for training translation models. For instance, (Kunchukuttan et al., 2018) developed the IIT Bombay Hindi-English corpus, a significant resource for Hindi-English translation tasks. There have been attempts to develop corpus and build translation systems for specific regional languages in India such as (Post et al., 2013; Revanuru et al., 2017; Laskar et al., 2019, 2020; Pathak et al., 2019; Pathak and Pakray, 2019; Choudhary et al., 2018; Singh et al., 2018). However, similar efforts for Maithili remain sparse.

The No Language Left Behind (NLLB) (Tiedemann, 2012) dataset is a part of Meta AI's NLLB initiative, which aims to improve machine translation for low-resource languages. The dataset includes parallel text for 200+ languages, including Maithili-Hindi. It is constructed from multiple sources, such as web-crawled data and publicly available datasets. There are 5,50,300 Maithili-Hindi parallel sentences available in OPUS[1] (Tiedemann, 2012; Fan et al., 2021; Schwenk et al., 2019). Furthermore, while the NLLB dataset provides a large number of Maithili-Hindi parallel sentences, its quality is poorer due to automatically generated translations and misalignments. In contrast, our dataset, although smaller (1,05,600 sentences), includes 5,600 manually verified sentences, and

[1] http://opus.nlpl.eu

the rest are synthetically generated. Further comparison of our data with NLLB is provided in the experiment section. Our contributions in the paper are as follows:

- We contribute a Maithili-Hindi parallel corpus comprising 1,05,600 sentences which includes 5,600 manually verified sentences.
- We fine-tune the SOTA MT models to present a strong baseline and show the superior quality of our data compared to the NLLB dataset.

2 Corpus Creation Methodology

We construct our corpora by using web scraping and optical character recognition (OCR) techniques. Data is sourced from various online repositories and printed materials, with different domains as detailed in Table 1. Web scraping is done on four websites: khattarkaka, videhamaithili, pranawjha.blogs, and maithilijindabaad (see Appendix A.1). Additionally, OCR is used for 141 books selected from the Maithili books collection. This section outlines the steps involved in creating the Maithili monolingual corpus and the Maithili-Hindi parallel dataset, as illustrated in Figure 1.

2.1 Book Digitization for Corpus Development

The Maithili data is collected from PDF files of various genre books like stories, conversations, and articles for our Maithili to Hindi MT task. We use Python libraries to extract the text and then process it. Specifically, we extract Maithili text from a PDF using Tesseract OCR (pytesseract)[2] in Python; this process involves converting PDF pages to images and then applying OCR to extract text from those images. To our understanding, Tesseract does not have a dedicated Maithili language model. However, Maithili uses the Devanagari script, which is supported by Tesseract's Hindi (hin) language data. This allows Tesseract to recognize Maithili text using the Hindi model.

2.2 Automated Web Scraping

Web scraping, the process of extracting data from websites, presents unique challenges, particularly when dealing with content in the Maithili language. One major challenge involves extracting specific HTML tags, such as '<h3>' for headings and ' ' for links, then systematically looping through these tags to gather the necessary information. We utilize the beautifulsoup[3] library to parse HTML and XML documents through parse trees to scrape out particular elements. Properly ordering and parsing nested HTML tags is complex, especially when transitioning between tags. Selenium is integrated to automate web browsing tasks, including page loading, link navigation, and handling dynamically loaded content to address the impracticality of manual navigation through genre and news pages.

S.N	Sources	Domain
1	khattarkaka	story, novel, satire
2	videhamaithili	literature, culture, history, society
3	pranawjha.blogs	articles, story
4	maithilijindabaad	literature, philosophy, culture, heritage, news
5	maithili-books	literature, story, history, culture.

Table 1: List of resources used to extract the monolingual Maithili corpus. More details in the Appendix Sec. A.1.

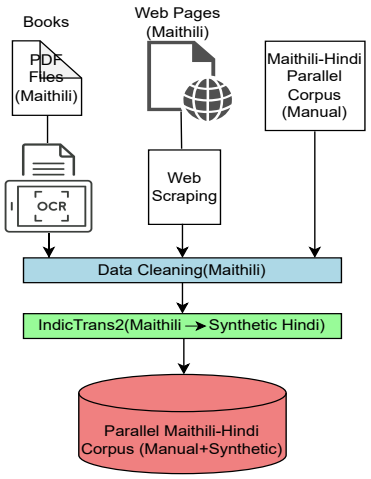

Figure 1: Detailed workflow for creating the Maithili monolingual corpus and Maithili-Hindi parallel dataset.

2.3 Data Cleaning

Once we extract the text, regex scripts are employed for text processing to remove English text and format the Maithili content. This involves removing unnecessary characters or symbols, normalizing the text (removing unnecessary, redundant punctuation marks, non-ASCII characters, and ex-

[2] https://github.com/madmaze/pytesseract

[3] https://pypi.org/project/beautifulsoup4

Dataset	Sentences	LaBSE	LASER2	Median	Standard deviation
Manually Created	5,600	**0.6925**	**0.7265**	**0.7129**	**0.1660**
Pseudo-Parallel	1,00,000	0.6678	0.4815	0.6952	0.1927
Combined (Manually + Pseudo)	1,05,600	0.6691	0.5026	0.6963	0.1915
NLLB	5,50,300	0.6659	0.3779	0.6958	0.2086

Table 2: Analysis of Avg. LaBSE, LASER, median similarity, and standard deviation across the Maithili-Hindi dataset

tra spaces), and segmenting the text into sentences or smaller units. The purpose of writing regex code is to clean the data as much as possible and make it structured. The cleaned Maithili data (1,00,000 sentences) is then stored in a text file format.

2.4 Manual and Pseudo data Generation

For manual translation, we gathered 5,600 Maithili texts from khattarkak and pranawjha.blogs and reviewed them thoroughly. Two linguistic experts, a 48-year-old male with qualifications of Ph.D and a 40-year-old male with qualifications of Master of Arts (Translation Studies), translated the 5,600 Maithili text sentence by sentence, ensuring accuracy and coherence.

For pseudo data generation, we use data augmentation (Sennrich et al., 2016) techniques to address the limited parallel corpora for Maithili-Hindi. IndicTrans2 (Gala et al., 2023) is used to generate synthetic parallel data by translating a 1,00,000 Maithili monolingual corpus into Hindi. These synthetic sentences are paired with their original Maithili counterparts to create additional parallel sentence pairs. The overall 1,05,600 sentences increase the training data size, exposing the NMT models to more diverse sentence structures and vocabulary. Compared to NLLB data, MaitH 1.0 has longer sentences on average (refer to Tables 5 and 6 in Appendix A.2).

2.5 Quality Check

To assess the quality and alignment of our parallel datasets, we employ two SOTA multilingual sentence embedding models: Language-agnostic BERT Sentence Embedding (LaBSE) (Feng et al., 2022) and LASER. These models project sentences from different languages into a shared semantic space, enabling direct comparison through cosine similarity. Table 2 provides a comparative analysis, showcasing the average similarity scores and variance for each dataset using both LaBSE and LASER.

From the results in Table 2, the manually curated parallel corpus exhibits the highest average similarity and the lowest standard deviation, indicating consistently strong alignment and minimal noise. In contrast, pseudo-parallel and NLLB data show comparatively lower average similarity and higher variability, reflecting weaker alignment and greater heterogeneity. We note that pseudo-parallel data has not been validated manually, possibly due to its large size and the availability of an expert in the Maitahli language. However, we show, in the experiment, that the baseline models achieve the highest performance on the *combined data* compared to manually created data alone, indicating the value of pseudo-parallel data. These findings underscore the importance of high-quality human-aligned data for building robust multilingual models and also provide an objective basis for selecting or filtering parallel corpora for low-resource machine translation tasks.

3 Experiments

This section presents data preprocessing, baseline models, results, and discussions. For the experiment, all the datasets are divided in the ratio of 80/10/10 for train/valid/test unless otherwise mentioned.

3.1 Data Preprocessing

The raw data often contains inconsistencies in text formatting, including varying Unicode encodings and the use of non-standard characters. We standardize the text by converting all characters to their normalized forms using Unicode normalization and apply the standard IndicNLP normalization (Kunchukuttan, 2020) to the corpus. The pretrained SentencePiece Model (SPM) (Gala et al., 2023) is used for subword tokenization (Kudo and Richardson, 2018). SentencePiece is an unsupervised subword tokenizer that efficiently handles the morphological richness of Maithili and Hindi. The final dictionaries for Maithili and Hindi comprised 1,22,706 and 1,22,672 unique subword units,

Model	BLEU4	chrF2	TER	COMET	METEOR	BERTScore
IndicTrans2	4.01	21.54	0.97	0.4365	0.2036	0.8835
mT5	26.56	54.62	0.58	0.6903	0.5000	0.9354
mBART50	23.82	52.90	0.61	0.6740	0.4850	0.9306
NLLB-200	**28.57**	**57.15**	**0.55**	**0.7292**	**0.5277**	**0.9387**

Table 3: Results of the models train and test on the manually created data

Model	Training Data	BLEU4	chrF2	TER	COMET	METEOR	BERTScore
IndicTrans2	Our	9.60	32.91	0.86	0.5248	0.3206	0.8951
	NLLB	2.12	16.41	1.40	0.4291	0.1353	0.8631
mT5	Our	15.42	47.38	0.71	0.5814	0.4614	0.9142
	NLLB	10.44	34.61	1.01	0.5679	0.2646	0.8835
mBART50	Our	25.94	52.85	0.63	0.6711	0.4865	0.9223
	NLLB	8.12	28.95	1.26	0.5181	0.1895	0.8732
NLLB-200	Our	**37.97**	**59.90**	**0.55**	**0.7356**	**0.5644**	**0.9382**
	NLLB	16.34	40.57	0.90	0.6092	0.3043	0.8959

Table 4: Each model is trained separately on our (MaitH 1.0) dataset and the NLLB dataset, and evaluated on the MaitH 1.0 test set.

respectively (See Appendix A.2 for details).

3.2 Baseline Models

We finetune four pre-trained multilingual models as baselines on MaitH 1.0 and NLLB training dataset: IndicTrans2 (Gala et al., 2023), mT5 (Xue et al., 2021), mBART50 (Liu et al., 2020), and NLLB-200 distilled model[4]. Each model is trained using task-specific hyperparameter configurations tailored for low-resource neural machine translation. To ensure clarity and reproducibility, the complete details of the hyperparameters used for finetuning the models are provided in Appendix A.3.

3.3 Evaluation Metrics

We report well-known BLEU4, character-level precision-recall F-score (ChrF2), Crosslingual Optimized Metric for Evaluation of Translation (COMET), METEOR, BERTScore, and Translation EDIT Rate (TER) metrics. The higher is the better for the first five metrics, whereas a lower value for TER is preferred. More details of the metrics are given in the Appendix A.4.

3.4 Results and Discussions

Experimental results are presented in Tables 4. We can observe that NLLB-200 outperforms the other three models on all metrics, likely due to it being pretrained on a massive amount of *parallel data* and a large number of languages, helping cross-lingual transferability.

Comparing metrics in Tables 4, we can see that models trained on the MaitH 1.0 consistently outperform models trained on the NLLB training data. It shows the superior quality of the MaitH 1.0 dataset. The results of all four models on manually curated 5600 samples are shown in Table 3, where the NLLB-200 model again outperforms others. Comparing the results in Tables 3 and 4, we can see that pseudo-parallel data further improves the performance metrics, thus supporting the value addition by pseudo-parallel data. Sample outputs are shown in Appendix A.5. The code, hyperparameters, and instructions for reproducing our results are provided in Appendix A.6.

4 Conclusion and Future works

Our work contributes a manually curated and synthetically generated parallel corpus for the Maithali-Hindi language pair. We also develop a strong baseline for Maithili-Hindi translation using our dataset. The study reveals the value of manually created and validated data (compared against NLLB, which is noisy). Future work will focus on improving synthetic data quality and incorporating domain-specific data. Additionally, fine-tuning models or new architecture with Maithili and Hindi-specific linguistic features may enhance their capabilities.

[4] https://huggingface.co/facebook/nllb-200-distilled-600M

5 Limitations

In this study, despite its valuable insights, it faces limitations due to limited manually curated data and reliance on synthetic data. The small dataset size and potential noise in synthetic data hinder model performance. More robust validation is needed for synthetic data to improve its quality. Improved training procedure in a low-data regime may help address these limitations and improve translation accuracy and fluency.

6 Ethical considerations

This research on Maithili-Hindi machine translation adhered to ethical principles, including data privacy and consent, bias and fairness, impact on low-resource languages, transparency and reproducibility, and avoidance of harm. Consent was obtained for manually curated data, and efforts were made to minimize bias in the models. The study aims to support the Maithili language community and was conducted transparently to ensure reproducibility. The research was designed to avoid any harm to individuals or communities. The dataset will be released under the CC-BY 4.0 license.

References

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.

Himanshu Choudhary, Aditya Kumar Pathak, Rajiv Ratan Saha, and Ponnurangam Kumaraguru. 2018. Neural machine translation for English-Tamil. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 770–775, Belgium, Brussels. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

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.

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

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

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.

Anoop Kunchukuttan. 2020. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The iit bombay english-hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Sahinur Rahman Laskar, Abinash Dutta, Partha Pakray, and Sivaji Bandyopadhyay. 2019. Neural machine translation: English to hindi. In *2019 IEEE Conference on Information and Communication Technology*, pages 1–6.

Sahinur Rahman Laskar, Abdullah Faiz Ur Rahman Khilji, Partha Pakray, and Sivaji Bandyopadhyay. 2020. Hindi-Marathi cross lingual model. In *Proceedings of the Fifth Conference on Machine Translation*, pages 396–401, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

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

Amarnath Pathak and Partha Pakray. 2019. Neural machine translation for indian languages. *Journal of Intelligent Systems*, 28(3):465–477.

Amarnath Pathak, Partha Pakray, and Jereemi Bentham. 2019. English–mizo machine translation using neural and statistical approaches. *Neural Comput. Appl.*, 31(11):7615–7631.

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.

Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. 2013. Improved speech-to-text translation with the fisher and callhome spanish–english speech translation corpus. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 138–148.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Karthik Revanuru, Kaushik Turlapaty, and Shrisha Rao. 2017. Neural machine translation of indian languages. In *Proceedings of the 10th Annual ACM India Compute Conference*, Compute '17, page 11–20, New York, NY, USA. Association for Computing Machinery.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2019. Ccmatrix: Mining billions of high-quality parallel sentences on the web. *arXiv preprint arXiv:1911.04944*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Shivkaran Singh, M. Anand Kumar, K.P. Soman, Sabu M. Thampi, El-Sayed M. El-Alfy, Sushmita Mitra, and Ljiljana Trajkovic. 2018. Attention based english to punjabi neural machine translation. *J. Intell. Fuzzy Syst.*, 34(3):1551–1559.

Matthew Snover, Bonnie Dorr, Richard 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.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675.

A Appendix

A.1 Dataset Resources

The following resources are used to collect and process the Maithili-Hindi parallel dataset:

- **Khattarkaka**: https://khattarkaka.com
- **Videha**: https://videhamaithili.wordpress.com/
- **Pranav Jha's Blog**: http://pranawjha.blogspot.com/
- **Maithili Jindabaad**: https://maithilijindabaad.com/
- **Archive.org (432 Maithili books)**: https://archive.org/details/432-MAITHILI-BOOKS

A.2 Maithili-Hindi parallel dataset Statistics

Table 5 presents the dataset statistics, including the number of sentences, tokens, type-token ratio (TTR), percentage of tokens replaced by <unk>, and average sentence lengths (in tokens) for the Maithili and Hindi datasets across the train and development splits. Statistics of the same language pair available in the NLLB dataset are shown in Table 6.

A.3 Hyperparameter Settings for Model Training

We finetune the IndicTrans2 custom 12-layer transformer with 512 embedding dimensions using Adam optimizer (Kingma and Ba, 2014) ($\beta(0.9, 0.98)$), a 3e-5 learning rate, 0.1 label smoothing, and an inverse_sqrt scheduler with 2000 warmup updates. Training runs for 35 epochs on an NVIDIA RTX A5000 GPU with a 0.2 dropout rate, gradient clipping (norm 1.0), mixed precision (fp16), and sequences are limited to 2,048 tokens.

mT5 (Xue et al., 2021) model comprises 12 encoder and 12 decoder layers, with 12 attention heads and an embedding dimension of 768. The

feed-forward network (FFN) dimension is set to 2048, using a GeGLU activation function with a dropout rate of 0.1. The model trains for 7 epochs with a batch size of 4 on an NVIDIA RTX A4500 GPU.

In our experiment, we finetune the pretrained *mBART50* model (Liu et al., 2020) with 12 encoder and decoder layers, each comprising 16 attention heads and an embedding dimension of 1024. The feed-forward network (FFN) dimensions are set to 4096. A dropout of 0.1, and the activation function uses ReLU. The training runs for 7 epochs with a batch size of 6, conducted on the same machine as mT5.

The NLLB model is finetune using the Hugging Face transformers library on a Maithili–Hindi parallel corpus. It is initialized from a publicly available distilled NLLB-200 checkpoint provided by Meta AI. The model follows the M2M100ForConditionalGeneration architecture with 12 encoder and 12 decoder layers, 16 attention heads, a hidden size of 1024, and a feed-forward dimension of 4096. All input and output sequences are truncated and padded to 128 tokens. The model is trained on an NVIDIA RTX A5000 GPU for 5 epochs with a learning rate of 2e-5, batch size of 8, and weight decay of 0.01. Mixed-precision (FP16) training is enabled, and the best checkpoint is selected based on the lowest evaluation loss.

A.4 Evaluation Metrics

To evaluate our Maithili to Hindi translation model, we use evaluation metrics commonly use in machine translation tasks. The most popular ones include BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002): Measures number of n-grams match between translations and reference texts with n=4, chrF (Character n-gram F-score) (Popović, 2015): Measures similarity using character n-grams (n=2), making it more effective for morphologically rich languages, TER (Translation Edit Rate) (Snover et al., 2006): Computes the minimum number of edits needed to convert a translation into the reference, and COMET (Rei et al., 2020):it is a neural-based translation evaluation metric that leverages pretrained transformer embeddings to assess translation quality. Unlike traditional metrics like BLEU, COMET is trained using direct human assessment (DA) ratings, making it more aligned with human judgments. It evaluates translations based on adequacy and fluency, given a source sentence, machine-generated translation, and reference translation. BERTScore (Zhang et al., 2019) computes a similarity score for each token in the candidate sentence with each token in the reference sentence. However, instead of exact matches, we compute token similarity using contextual embeddings. METEOR (Banerjee and Lavie, 2005) evaluates a translation by computing a score based on ex- plicit word-to-word matches between the transla- tion and a reference translation.

We use sacrebleu[5] library to compute BLEU and chrF scores, pyter[6] library to compute TER score, and Unbabel[7] library to compute COMET, bert_score[8] library to compute BERTScore, meteor_score library is use to compute METEOR, which is a standard for evaluating machine translation outputs.

A.5 Model Output on Test Samples

To analyze the performance of our fine-tuned models, we present sample translations from our Maithili-Hindi test dataset. The figure 2 below showcases translations generated by IndicTrans2, mBART50, mT5, and NLLB-200, alongside the original Maithili sentence and the reference Hindi translation. The comparison highlights the differences in translation quality across models.

A.6 Code and Reproducability

To support reproducibility and further research in Maithili-Hindi machine translation, we plan to publicly release our Maithili-Hindi parallel dataset upon the acceptance of this paper. The dataset and code will be made available at https://anonymous.4open.science/r/anonymous-mt-data_and_code-CD2F/

[5] https://pypi.org/project/sacrebleu
[6] https://pypi.org/project/pyter3
[7] https://github.com/Unbabel/COMET
[8] https://github.com/Tiiiger/bert_score

Source 1	हमरा मुँह सँ किछु उत्तर नहि बहराएल ।
Reference translation	मेरे मुँह से कुछ उत्तर नहीं निकाला।
Gloss:	No answer came out of my mouth.
IndicTrans2	मुझे मुँह से कुछ जवाब नहीं मिला।
Gloss:	I did not get any reply from my mouth.
mT5	मेरे मुँह से कुछ उत्तर नहीं बहराया।
Gloss:	No reply came out of my mouth.
mBART50	मेरे मुँह से कुछ भी उत्तर नहीं निकला।
Gloss:	No reply came out of my mouth.
NLLB-200	मेरे मुँह से कोई जवाब नहीं निकला।
Gloss:	No answer came out of my mouth.
Source 2	आब जा कऽ बड़की बाबी ओ सहजोपीसी कैं वस्तुस्थितिक बोध भेलैन्ह ।
Reference translation	अब जाकर बड़ी दादी और सहजो बुआ को वस्तुस्थिति का बोध हुआ।
Gloss:	Now elder grandmother and Sahajo Bua realized the reality.
IndicTrans2	अब जब बड़ी बड़ी दादी और सहार को वस्तु की वस्तुएँ हुई थीं।
Gloss:	Now when the elder grandmother and Sahar had become objects of the matter.
mT5	अब जाकर बड़ी दादी और सहजोपीसी को वस्तुस्थिति का बोध हुआ।
Gloss:	Now the elder grandmother and sister-in-law realized the true situation.
mBART50	अब जाकर बड़ी दादी और सहजो बुआ को वस्तुस्थिति का बोध हुआ।
Gloss:	Now elder grandmother and Sahajo Bua realized the reality.
NLLB-200	अब जाकर बड़ी दादी और सहजोपीसी को वस्तुस्थिति का एहसास हुआ।
Gloss:	It was only now that the elder grandmother and Sahajopisi realized the factual situation.
Source 3	हम चुपचाप अपन सूटकेस ओ विस्तर उठाओल और रिक्सापर चढि पराजित सैनिक जकाँ स्टेशन विदा भेलहुँ।
Reference translation	मैं चुपचाप अपना सूटकेस और बिस्तर उठाया और रिक्शा पर चढ़ कर पराजित सैनिक जैसे स्टेशन विदा हुआ ।
Gloss:	I silently picked up my suitcase and bedding, boarded a rickshaw and left the station like a defeated soldier.
IndicTrans2	मैं चुपचाप चुपचाप अपने तीरों को उठा और दरवाजे पर चढ़ाई की तरह चढ़ गया।
Gloss:	I quietly and silently picked up my arrows and climbed like a ladder to the door.
mT5	मैं चुपचाप अपना सूटकेस और विस्तर उठाया और रिक्शा पर चढ़कर पराजित सैनिक की तरह स्टेशन चला गया।
Gloss:	I silently picked up my suitcase and bedding, boarded a rickshaw and went to the station like a defeated soldier.
mBART50	मैंने चुपचाप अपना सूटकेस ओ विस्तर उठाया और रिक्सा पर चढ़कर पराजित सैनिकों के जैसे स्टेशन से विदा हुआ।
Gloss:	I silently picked up my suitcase and bedding, boarded a rickshaw and left the station like a defeated soldier.
NLLB-200	मैंने चुपचाप अपना सूटकेस और चौड़ा उठाया और रिक्शे पर चढ़कर पराजित सिपाही की तरह स्टेशन से रवाना हो गया।
Gloss:	I quietly lifted my suitcase wider and climbed into the rickshaw and started from the station like a defeated soldier.
Source 4	नदी त अछिए नै! गामक बलान नदी सुखि क पीच रोड बनल अछि आ साइकिल, मोटरसाईकिल सभ ओई बाटे सरसरायल ऐ पार से ओइ पार भ रहल अछि।
Reference translation	नदी तो है ही नहीं! गांव का बलान नदी सूखकर पिच रोड बना हुआ है, और साइकिल, मोटरसाइकिल, उस रास्ते से सरसराते हुए इस पार से उसे पार हो रहे हैं।
Gloss:	There is no river! The Balan river of the village has dried up and has become a paved road, and bicycles and motorcycles are swooshing from one side to the other.
IndicTrans2	नदी तो नहीं है! गाँव के बल नदी के किनारे बने हुए हैं, साइकिल और साइकिल के पार के पार से सभी पार हो रहे हैं।
Gloss:	There is no river! The village's boats are built on the river bank, everyone is crossing it on bicycles and bikes.
mT5	नदी तो है ही नहीं! गाँव की बलान नदी सूखी पीच रोड बना हुआ है और साइकिल, मोटरसाइकिल सब उस रास्ते से सरसराए इस पार से ओइ पार हो रही है।
Gloss:	There is no river! The Balan river in the village is a dry tar road and bicycles and motorcycles are crossing it from one side to the other.
mBART50	नदी तो है ही नहीं! गाँव का बल नदी सुखकर पीच रोड बना हुआ है और साइकिल, मोटरसाइकिल सब उसी रास्ते से सरसराए हुए इस तरह से इस तरह से सरसराए जा रहे हैं।
Gloss:	There is no river! The riverbed of the village has dried up and turned into a tar road and bicycles, motorcycles, everyone is moving along that path in this way and that way.
NLLB-200	नदी तो है ही नहीं! गाँव की बलान नदी सुखी का पीच रोड बना हुआ है और साइकिल, मोटरसाइकिल सभी उसी रास्ते सरसरायल से उसी रास्ते से उसी रास्ते से गुजर रही है।
Gloss:	There is no river! The village's Balan river is a pitch road and bicycles, motorcycles, all pass through the same route from Sarasrail.

Figure 2: Output of the finetuned model, Maithili (source) text to Hindi (Target) translations from our test dataset.

Dataset	Language	Sentences	Tokens	TTR	Replaced by <unk>	Avg Sentence Length
Train	Maithili	84,480	2,192,627	0.0559	0.127%	16.25
Train	Hindi	84,480	2,130,752	0.0576	0.0023%	19.19
Dev	Maithili	10,560	2,69,184	0.4558	0.0858%	15.79
Dev	Hindi	10,560	2,62,342	0.4562	0.00191%	19.15
Test	Maithili	10,560	2,53,779	0.4835	0.0686%	15.90
Test	Hindi	10,560	2,36,349	0.5190	0.000846%	18.60

Table 5: Statistics of MaitH 1.0 Maithili-Hindi parallel dataset

Dataset	Language	Sentences	Tokens	TTR	Replaced by <unk>	Avg Sentence Length
Train	Maithili	4,40,240	53,28,862	0.0230	0.02%	6.53
Train	Hindi	4,40,240	37,45,318	0.0327	0.409 %	5.78
Dev	Maithili	55,030	6,82,981	0.1796	0.0171%	6.66
Dev	Hindi	55,030	4,70,957	0.2604	0.43 %	5.77
Test	Maithili	55,030	6,87,058	0.1785	0.0192 %	6.69
Test	Hindi	55,030	4,73,530	0.2590	0.42%	5.79

Table 6: Statistics of existing NLLB Maithili-Hindi parallel dataset