LLMs for Translation of Low-Resource Languages: An India-centric study

Chinmay Jain & Lathiya Sahil & Dharwada Sesha Sriram
Indian Institute of Science
Bengaluru, KA, India
chinmayjain@iisc.ac.in, sahilm@iisc.ac.in, sriramd@iisc.ac.in

1 Abstract

Vast linguistic diversity of India presents difficulties in accessing online content due to lack of Indians' proficiency in English. Machine translation is a solution, but challenges such as scarcity of high quality parallel corpus, structural disparities between English and Indian languages are faced. LLMs like ChatGPT offer potential in MT, but their effectiveness varies and hasn't been studied in detail for Indic languages. Our research focuses on refining LLMs' translation quality, particularly for North Eastern Indian languages.

2 Introduction

India is linguistically a very diverse country. It has 23 languages recognised officially and many unpopular languages with different dialects. However, only 11% of Indians are proficient in English. Since the information and data on the internet is majorly available in English, this poses a major challenge for the Indians who are not comfortable with English to access the internet's knowledge and various resources.

As a result, a critical necessity emerges to translate this valuable information into local languages and enable knowledge sharing among the Indian population. Language translation can also play a crucial role in exchanging feelings, opinions and actions, enabling better communication and understanding among different linguistic groups. In this digital era reducing this communication gap can even open up more economic opportunities in terms of trade, tourism, commerce among different linguistic regions of India.

But manually translating is not practically feasible as it would be very time consuming, and expensive. A better and viable solution is machine translation. Machine translation has its own set of problems of which unavailability of high quality parallel corpora is the biggest issue. And there are many other boulders like structural differences, morphological complexity, word order differences, lexical gaps, variations due to dialects between English and Indian languages. All of these have complicated the translation process furthermore.

The arrival of large language models (LLMs) like ChatGPT, LLaMA, and PaLM has significantly propelled the field of machine translation (MT) in recent times. Not only can LLMs be used in various scenarios beyond MT, such as Question Answering, Summarization, and Classification, the zero-shot MT performance of LLMs is comparable with strong fully supervised MT, whereas the few-shot MT performance of LLMs even surpasses fully supervised MT models in some cases. However, the translation quality of LLMs is known to be worse when translating out of English compared to translating into English. In addition, very limited work has been done in exploring LLMs for MT focused on Indic Languages.

This gap in research presents an excellent opportunity to explore the MT capabilities of LLMs for Indian Languages and narrow their performance gap in translating in and out of English. Also given the need of developing MT systems with limited parallel datasets, in our work, we wish to evaluate different prompting and training techniques of LLMs in the low-resource translation task of Indic languages. Investigating the capacity of LLMs to produce rich translations remains an open challenge, yet it is a valuable pursuit that can facilitate widespread and easy access to proficient MT systems for Indian languages.

Although our initial focus was on translating North Eastern languages of India since they are very low in resource, preliminary experiments revealed that LLMs have extremely little to no prior knowledge of these languages. In fact, we observed that their comprehension of Hindi, the most widely spoken and documented Indian language, is also quite limited. The pretraining phase of the most popular open source LLM, LLaMA, does not use data from any Indic languages at all. Hence, given Hindi's ease of interpretability and its abundance of parallel corpora, we decided to shift our focus to translating Hindi to and from English.

Our contributions in this work are the following:

- Extensive zero-shot bench-marking done on IIT Bombay English-Hindi Prallel Corpus and state-of-the-art open-sorce LLMs LLaMA, Mistral, Gemma, and OpenHathi.
- Benchmarking with respect to different prompting strategies 0-shot, 2-shot, and 4-shot prompting.
- Initial stages of fine-tuning LLMs with QLoRA.

Once we develop a robust Hindi translation model, we would then apply the same training scheme to India's North Eastern languages.

3 Related Work

Transformer Models for Low-Resource Indic Translation: Previous work has explored low-resource translation for Indic languages thanks to WMT's shared task in 2023. However all the submissions under that shared task have employed only transformer based models. A brief summary of notable submissions is presented below.

Agrawal et al, 2023 focus on machine translation of Assamese and Manipuri from and to English. Their base model is the transformer based IndicTrans2, which is pre-trained on the Bharat Parallel Corpus Collection (BPCC). Since scarcity of high quality parallel corpus is a major difficulty faced in low-resource machine translation, the authors fine-tuned IndicTrans2 using transfer learning techniques on English-Assamese and English-Manipuri pairs to capaitalize on the model's existing knowledge. They produced excellent results with BLEU scores 47.54 for MNI→EN 47.54, 26.36 for EN→MNI, 35.24 for ASM→EN, and 18.15 for EN→ASM.

Dabre et al. (2023) implement a three-step strategy, integrating denoising and MT training, back translation, and parallel corpus training, boosting translation quality by up to 4 BLEU points. Employing fine-tuning and system combination, their approach significantly enhances performance for specific language pairs in low-resource settings, exemplified by notable BLEU scores: MNI→EN: 43.35, EN→MNI: 27.40, ASM→EN: 36.97, EN→ASM: 21.07, MZ→EN: 33.30, EN→MZ: 33.64.

LLM's for Machine Translation: GPT-3 (Brown et al., 2020) showcased the capability of large language models to perform few-shot predictions with only a simple description of the task in natural language, and optionally, a few examples showing how the task must be performed. Without any gradient updates or fine-tuning, GPT-3 in the few-shot setting significantly outperformed prior unsupervised machine translation models when translating into English but underperformed in the opposite direction due to tokenization issues.

Following GPT-3, large language models have emerged in abundance, further advancing the state of the art. PaLM (Chowdhery et al., 2022), in particular, continued this trajectory of improving language modeling capabilities, training a transformer with 540 billion parameters on a corpus of 780 billion text tokens. It surpasses state-of-the-art language models in translation tasks between English and French, German, and Romanian without explicit parallel text training. It excels in 0-shot, 1-shot, and few-shot scenarios but struggles with extremely low-resource languages and translating from non-English languages.

An important observation from both GPT-3 and PaLM is that the evaluations of few-shot translation beats that of 0-shot translation very decisively when the source or target language is English. This difference is in the range of 5.1-19.6 BLEU when translating with GPT-3, and 1.7-5.6 BLEU when translating with PaLM.

Instead of simply prompting to evoke quality translations, recent work has shown that by applying a novel fine-tuning strategy, LLMs can be trained for machine translation with very less parallel data and comparatively smaller computational cost. Xu et al., 2023 propose two stages of fine-tuning LLMs to achieve this. In the first stage, the focus lies on refining the LLMs' proficiency in non-English languages relevant to the translation task through fine-tuning on monolingual data of those languages. In the second stage, the emphasis shifts towards translation generation by fine-tuning the LLMs on a small yet high quality parallel dataset. Remarkably, the performance of LLaMA-2-7B when translating from English shoots up from 13.86 BLEU in the 0-shot setting, to 29.78 BLEU after two stages of fine-tuning.

No literature survey on machine translation for Indic languages can be complete without covering AI4Bharat's IndicTrans2 (Gala et al., 2023), an incredible contribution to the field, in detail. IndicTrans2 is the first open-source transformer-based multilingual NMT model that supports high-quality translations across all the 22 scheduled Indic languages. Gala et al., 2023 is also the first and only work that has benchmarked an LLM model (GPT3.5) on the Indic languages translation task in both the En-Indic and Indic-En directions. They observe that LLMs show promising zero-shot capabilities but still lag behind the task-specific models, particularly for low-resource languages.

4 Models Evaluated:

Restricting to open source models, we evaluated the performance of the following LLMs.

1. LLaMA 2 [Touvron, Hugo, et al.] family of pretrained and fine-tuned LLM's released by Meta AI, available in two main types: LLaMA 2 Base, and LLaMA 2 Chat. Both of these models are available in three versions with 7B, 13B and 70B parameters each offering a balance between size/performance and the quality of responses. The models are trained on 2.2T tokens with 4096 token context length, uses RMSprop, Rotary positional embedding, and SwiGLU activation techniques.

2. Mistral[Jiang, Albert Q., et al.]: A 7-billion parameter decoder-based LLM. Its achitectural choices include sliding window attention, grouped query attention, and a byte-fallback BPE tokenizer. These enable Mistral to handle longer sequences, perform faster inference, and tackle a broader character range compared to traditional models.

3. Gemma 7B : LLM by Google AI, offered in two variants having 2B and 7B parameters. It is built upon the transformer decoder framework with Multi-Query Attention,RoPE Embedding, and GeGLU Activations.

4. OpenHathi developed by AI4Bharat and SarvamAI, is designed specifically to excel in Indic languages, particularly Hindi. The model released is OpenHathi-7B-Hi, which was trained on Hindi, English, and Hinglish. It leverages the sub-word tokenisation rest of the architectural details are yet to be disclosed.

5 Prompting Techniques:

There are two primary prompting techniques:

1. Zero-Shot Prompting: A technique in which models are instructed to perform tasks for which it was not specifically trained for. In simple words, it is prompting without giving relevant examples. This approach tests the model's inherent ability to understand the task.

2. Few Shot Prompting: Taking the zero shot to a step further, the model is provided with few examples for a specific task along with the prompt. The aim is to enhance the response of model. In a way, we are fine tuning the model by presenting training data in prompt itself.

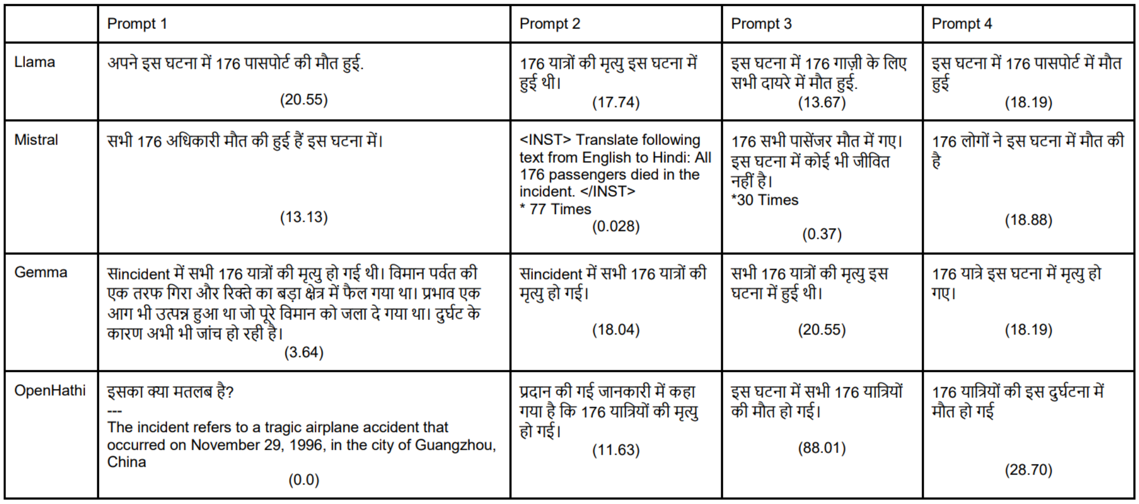
We observe that few shot learning improves significantly in accuracy, sentence structure of translation and response format, style compared to zero shot learning. The improved response comes at the cost of computational resources. Effectiveness of prompting depends largely on the template of prompt used. We tested the following zero-shot prompt templates on different models:

Prompt 1: Translate the following text from English to Hindi:
$English$
$hindi$

Prompt 2: $<s><INST>$ Translate following text from English to $Hindi:$
$English</INST>$
$hindi$

Prompt 3: Translate from English to Hindi no extra text
$[English]:english$
$[Hindi]:$

Prompt 4: System: You are a Translator from English to Hindi :
User: What is the Translation of the sentence given below.
$[English]:english$
$[Hindi]:$



	Prompt 1	Prompt 2	Prompt 3	Prompt 4
Llama	अपने इस घटना में 176 पासपोर्ट की मौत हुई. (20.55)	176 यात्रों की मृत्यु इस घटना में हुई थी। (17.74)	इस घटना में 176 गाज़ी के लिए सभी दायरे में मौत हुई. (13.67)	इस घटना में 176 पासपोर्ट में मौत हुई (18.19)
Mistral	सभी 176 अधिकारी मौत की हुई हैं इस घटना में। (13.13)	<INST> Translate following text from English to Hindi: All 176 passengers died in the incident. </INST> * 77 Times (0.028)	176 सभी पासेंजर मौत में गए। इस घटना में कोई भी जीवित नहीं है। *30 Times (0.37)	176 लोगों ने इस घटना में मौत की है (18.88)
Gemma	Sइincident में सभी 176 यात्रों की मृत्यु हो गई थी। विमान पर्वत की एक तरफ गिरा और रिक्ते का बड़ा क्षेत्र में फैल गया था। प्रभाव एक आग भी उत्पन्न हुआ था जो पूरे विमान को जला दे गया था। दुर्घट के कारण अभी भी जांच हो रही है। (3.64)	Sइincident में सभी 176 यात्रों की मृत्यु हो गई। (18.04)	सभी 176 यात्रों की मृत्यु इस घटना में हुई थी। (20.55)	176 यात्रे इस घटना में मृत्यु हो गए। (18.19)
OpenHathi	इसका क्या मतलब है? --- The incident refers to a tragic airplane accident that occurred on November 29, 1996, in the city of Guangzhou, China (0.0)	प्रदान की गई जानकारी में कहा गया है कि 176 यात्रियों की मृत्यु हो गई। (11.63)	इस घटना में सभी 176 यात्रियों की मौत हो गई। (88.01)	176 यात्रियों की इस दुर्घटना में मौत हो गई (28.70)

Actual English Sentence: All 176 passengers died in the incident
Bleu score are mentioned for each sentence in paranthesis

Prompt 4 gave us the most consistent results, so we performed experiments using Prompt 4.
The prompt used for Two-Shot prompting technique is :
$System:$ You are a Translator from English to Hindi:
$User:$
$Example1:$
$[English]:english1$
$[Hindi]:hindi1$
$Example2:$
$[English]:english2$
$[Hindi]:hindi2$
Following the same format above from the examples, What is the Translation of the sentence given below.
$[English]:english$
$[Hindi]:$
Similarly we added more examples for Four-Shot Prompting.

6 Fine-Tuning LLMs:

Fine-Tuning allows us to leverage existing pre-trained foundation models and adapt them to specific tasks or domains. By training the model on domain-specific data, we can tailor it to perform well on targeted tasks.
Fine tuning a LLM model is time consuming and resource intensive task which requires lots of training data and efforts and in this process the model might lost the previously acquired

knowledge from Pre-training also called as catastrophic forgetting.This is where PEFT and LoRA comes in.

1. PEFT (Parameter-Efficient Fine-Tuning):
   Unlike full fine-tuning (updating all parameters), PEFT efficiently adapts LLMs by focusing on task-relevant parameters identified through techniques like structured pruning. The reduced sized model is then fine-tuned for the specific task.
2. LoRA (Low-Rank Adaptation):LoRA improves LLM's performance by adjusting the layer relevance in the model architecture during fine-tuning. Unlike full fine tuning, it dynamically modulates each layer's contribution to the final output based on task requirements, improving adaptability without extensive tuning or architecture changes. This enhances the model's flexibility and versatility.

7 Evaluation Details:

1. Dataset: The datasets used were:
   (a) Training Dataset: We collected English-Hindi pairs for training from Samanantar. These pairs were used as examples for Two-Shot prompting, Four-Shot prompting, and fine-tuning.
   (b) Evaluation Dataset: Due to lack of a validation set in Samanantar we used the CFILT IITB Hindi English Corpus validation set.
2. Evaluation Metric: BLEU Score
   We used the BLEU score (Bilingual Evaluation Understudy) to evaluate the quality of translation.The score ranges from 0 to 100 ,higher score indicating better translation.Being most widely used metric or translation it allows to compare our findings with the SOTA implementations.
3. Hardware Infrastructure:
   We leveraged the computational power of two high-performance servers:
   (a) Wells Fargo server: Equipped with eight NVIDIA A100 GPU's, each having 40GB of dedicated RAM and 256 processors.
   (b) DGX1 server: Eight Tesla V100-SXM2 GPU's ,each having 32 GB GPU RAM and 40 processors.

8 Results:

The zero-shot evaluations on IIT-B Validation set by LLaMA and OpenHathi are as follows:

	English to Hindi	Hindi to English
LLaMA 7B-Base	5.577446144	6.97621922
LLaMA 7B-chat	5.064755656	11.32268463
OpenHathi	17.5976813	15.9491384

The two shot and four shot evaluations on IIT-B Validation set by LLaMA 7B-Base are as follows:

	English to Hindi	Hindi to English
2 shot	6.97621922	9.110884911
4 shot	8.317730645	8.816234475

The finetuning results of LLaMA 7B-Base on the English to Hindi task on the IIT-B Validation set are as follows:

Number of training examples	200	500	2500	10000
	7.21	7.64	8.34	8.14

All fine-tuning experiments were done with LoRA. The LoRA attension dimension is set to 64, the alpha parameter for LoRA scaling is set to 16, and the dropout probability for LoRA layers is set to 0.1. The models were trained for 2 epochs as they were starting to overfit after that point.

The best performance is achieved with fine-tuning on only 2500 examples. Upon further increasing the number of training examples to 10000, the model performance does not increase. Fine-tuning with 2500 examples is a clear improvement compared to Zero-Shot prompting by 3 BLEU points, but only slightly edges the performance of Four-Shot prompting on English to Hindi.

Model	BLEU	Pretraining	Training	# of Asm tokens
En-Ass Base 50k 2 epochs	3.37	None	50k En-Ass pairs	1.6M
En-Ass Base 450k 2 epochs	5.02	None	50k En-Ass pairs + 400k Backtranslated En-Ass pairs	1.6M + 16.4M
En-Ass Base 1.6M 2 epochs	7.24	None	50k En-Ass pairs + 1.55M Backtranslated En-Ass pairs	1.6M + 60.5M
En-Ass Instr tuned 50k 2 epochs	4.43	Assamese Alpaca & GPT Instr	50k En-Ass pairs	1.6M + 18M tokens

9 Next Steps

With our current hardware, it takes roughly 4 hours to perform complete BLEU score evaluation on the IITB-Validation set, and roughly 8 hours to fine-tune LLaMA 7B Base on 10000 examples. Due to this, we were limited in the number of experiments performed so far. The next set of experiments would be:
1. Perform Zero-Shot and Few-Shot evaluations using Gemma and Mistral 2. Fine-tune all the LLMs on both English to Hindi as well as Hindi to English tasks. 3. Experiment with multiple rounds of fine-tuning.

10 Contributions

Dharwada Sesha Sriram: Inferencing Mistral, Finetuning
Sahil Lathiya: Inferencing LLaMA2, openHathi, Finetuning
Chinmay Jain: Inferencing Gemma, few shots

Github Repository - Github link

References

[1] Gala, Jay, et al. ”IndicTrans2: Towards High-Quality and Accessible Machine Translation Models for all 22 Scheduled Indian Languages.” arXiv preprint arXiv:2305.16307 (2023).

[2] Zhu, W., Liu, H., Dong, Q., Xu, J., Huang, S., Kong, L., Chen, J., & Li, L. (2023). Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. [Preprint]. arXiv. Retrieved from https://arxiv.org/abs/2304.04675

[3] Agrawal, G., Das, R., Biswas, A., & Thounaojam, D. M. (2023). Neural Machine Translation for English - Manipuri and English - Assamese. In Proceedings of the

Eighth Conference on Machine Translation (pp. 931-934). Singapore: Association for Computational Linguistics. https://aclanthology.org/2023.wmt-1.86

[4] Xu, Haoran, et al. ”A paradigm shift in machine translation: Boosting translation performance of large language models.” arXiv preprint arXiv:2309.11674 (2023).

[5] NICT-AI4B’s Submission to the Indic MT Shared Task in WMT 2023 (https://aclanthology.org/2023.wmt-1.88) (Dabre et al., WMT 2023)

[6] Gaikwad, P., Doshi, M., Deoghare, S.D., & Bhattacharyya, P. (2023). Machine Translation Advancements for Low-Resource Indian Languages in WMT23: CFILT-IITB’s Effort for Bridging the Gap. Conference on Machine Translation.

[7] Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N.M., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B.C., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., García, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A.M., Pillai, T.S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Díaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K.S., Eck, D., Dean, J., Petrov, S., & Fiedel, N. (2022). PaLM: Scaling Language Modeling with Pathways. J. Mach. Learn. Res., 24, 240:1-240:113.

[8] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS’20). Curran Associates Inc., Red Hook, NY, USA, Article 159, 1877–1901.

[9] Hada, R., Gumma, V., de Wynter, A., Diddee, H., Ahmed, M., Choudhury, M., Bali, K., & Sitaram, S. (2024). Are Large Language Model-based Evaluators the Solution to Scaling Up Multilingual Evaluation? [Preprint]. arXiv. Retrieved from https://arxiv.org/abs/2309.07462

[10] Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4948–4961, Online. Association for Computational Linguistics.

[11] Candy Lalrempuii, April 4, 2023, ”LUS: Mizo Monolingual Corpus”, IEEE Dataport, doi: https://dx.doi.org/10.21227/5601-9c25.

[12] Signoroni, E., & Rychly, P. (2023). MUNI-NLP Systems for Low-resource Indic Machine Translation. In Proceedings of the Eighth Conference on Machine Translation (pp. 959-966). Association for Computational Linguistics. Singapore. Retrieved from https://aclanthology.org/2023.wmt-1.91

[13] https://www.theaidream.com/post/google-gemma-open-source-llm-everything-you-need-to-know

[14] Hu, Edward J., et al. ”Lora: Low-rank adaptation of large language models.” arXiv preprint arXiv:2106.09685 (2021).

[15] https://huggingface.co/docs/peft/main/en/conceptual_guides/lora

[16] https://abvijaykumar.medium.com/fine-tuning-llm-parameter-efficient-fine-tuning-peft-lora-qlora-part-1-571a472612c4

[17] Zhang, Biao, Barry Haddow, and Alexandra Birch. ”Prompting large language model for machine translation: A case study.” International Conference on Machine Learning. PMLR, 2023.

[18] Jiang, Albert Q., et al. ”Mistral 7B.” arXiv preprint arXiv:2310.06825 (2023).