Adapting Multilingual LLMs to Low-Resource Languages using Continued Pre-training and Synthetic Corpus: A Case Study for Hindi LLMs

Raviraj Joshi, Kanishk Singla, Anusha Kamath, Raunak Kalani, Rakesh Paul, Utkarsh Vaidya, Sanjay Singh Chauhan, Niranjan Wartikar, Eileen Long
NVIDIA
{ravirajj, kanishks, anushak, rkalani, rapaul, uvaidya
schauhan, nwartikar, elong}@nvidia.com

Abstract

Multilingual LLMs support a variety of languages; however, their performance is suboptimal for low-resource languages. In this work, we emphasize the importance of continued pre-training of multilingual LLMs and the use of translation-based synthetic pre-training corpora for improving LLMs in low-resource languages. We conduct our study in the context of the low-resource Indic language Hindi. We introduce Nemotron-Mini-Hindi 4B, a bilingual SLM supporting both Hindi and English, based on Nemotron-Mini 4B. The model is trained using a mix of real and synthetic Hindi + English tokens, with continuous pre-training performed on 400B tokens. We demonstrate that both the base and instruct models achieve state-of-the-art results on Hindi benchmarks while remaining competitive on English tasks. Additionally, we observe that the continued pre-training approach enhances the model's overall factual accuracy.

1 Introduction

The accuracy and utility of large language models (LLMs) have continuously improved over time. Both closed and open-source LLMs have demonstrated strong performance in English and several other languages. Open models such as Nemotron (Adler et al., 2024), Gemma (Team et al., 2024), and Llama (Dubey et al., 2024) are inherently multilingual. For instance, the Nemotron-4 15B model was pre-trained on 8 trillion tokens, of which 15% were multilingual (Parmar et al., 2024). However, the proportion of multilingual data is limited, which in turn affects the accuracy of these models on non-English languages.

The model's performance further diminishes as we move from high-resource to low-resource languages. In this work, we specifically focus on the Indic language Hindi as our target low-resource language. Out of the 8 trillion tokens used to train the Nemotron-4 models, only 20 billion tokens are in Hindi. As a result, while the model can understand and generate Hindi content to a reasonable extent, the usability of such a multilingual LLM for specific low-resource languages remains questionable. Frequent hallucinations, meaningless sentences, and mixing of English content often occur when responding to purely Hindi queries in the Devanagari script. There is a strong need to adapt multilingual LLMs to target languages to enhance their usability.

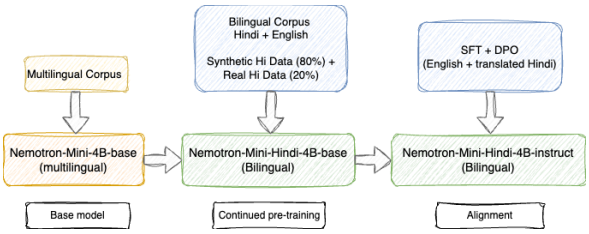

Figure 1: Adaptation of multilingual Nemotron-Mini-4B model (also known as Minitron-4B).

Recently, in the context of Indic languages, target language Supervised Fine-Tuning (SFT) has become a common practice to adapt LLMs to specific languages (Gala et al., 2024). However, it remains to be studied whether language-specific SFT tuning improves LLMs' understanding in regional contexts. Some studies suggest that SFT can introduce LLMs to new domain knowledge, though it is typically used to enhance the model's instruction-following capability (Mecklenburg et al., 2024). SFT on translated English instruction tuning data is widely used to develop regional LLMs for Indic languages. While this may improve instruction-following in the target language, it may not enhance LLMs' understanding of regional contexts (Balachandran, 2023). Another approach to updating LLM knowledge is continued pre-training, but the limited availability of tokens for low-resource languages makes this both infeasible and prone to

Proceedings of the First Workshop on Natural Language Processing for Indo-Aryan and Dravidian Languages (IndoNLP2025), pages 75–82
January 20, 2025.

Model	Layers	Hidden Size	Att. Heads	Query Groups	MLP Hidden	Parameters
Nemotron 4B	32	3072	24	8	9216	4.19B

Table 1: Architecture details of Nemotron-Mini-4B model.

overfitting.

In this work, we focus on a continued pre-training approach using a mix of real and synthetic corpora. We demonstrate that a robust base model can be adapted to the target language with a small continued pre-training corpus. This approach is particularly relevant for low-resource languages, where the amount of training data is limited. The synthetic pre-training dataset is curated by translating high-quality generic English corpora into the target language. To further expand the corpus and support Roman script queries in the target language, the text is transliterated into Roman script and used for pre-training. The base model is then aligned using supervised fine-tuning (SFT), followed by preference tuning with Direct Preference Optimization (DPO). We observe that the continued pre-training approach is particularly useful for reducing hallucinations, improving regional knowledge of LLMs, and enhancing response capabilities in the target language. The high-level process is outlined in Figure 1,

Based on this approach, we present Nemotron-Mini-Hindi-4B-Base[1] and Nemotron-Mini-Hindi-4B-Instruct[2][3], state-of-the-art Small Language Models (SLMs) for the Hindi language. These SLMs support Hindi, English, and Hinglish. The Hindi models are based on the multilingual Nemotron-Mini-4B (also known as Minitron-4B), adapted with continued pre-training on 400 billion Hindi and English tokens. The data blend used equal proportions of both languages. The instruct version of the model was developed using SFT and DPO techniques. The model outperforms all similarly sized models on various IndicXTREME, IndicNLG benchmark tasks and popular translated English benchmarks such as MMLU, Hellaswag, ARC-C, and ARC-E (Gala et al., 2024). We also perform LLM-based evaluations using the benchmark datasets IndicQuest (Rohera et al., 2024) and in-house SubjectiveEval, with GPT-4 serving as the judge LLM. This is the first study to present and evaluate bilingual language models of this nature. We provide a thorough study of the models in both languages.

2 Related Work

In this section, we review various approaches for adapting LLMs to different languages. Several efforts have focused on adapting LLaMA models to Indic languages. A common method involves extending the vocabulary, followed by SFT or PEFT (LoRA) using translated and available SFT corpora in Indic languages. Examples of such work include OpenHathi, Airavata (Gala et al., 2024), Tamil-LLaMA (Balachandran, 2023), Navarasa[4], Ambari, MalayaLLM, and Marathi-Gemma (Joshi, 2022). Notably, some of these efforts employ bilingual next-word prediction, alternating between English and the target language in the pre-training corpus. Airavata also introduced an evaluation framework[5] for Indic LLMs, which we leverage to evaluate Nemotron-Mini-Hindi 4B and other multilingual models.

Apart from Indic languages, similar efforts have been made for other languages, including Chinese LLaMA (Cui et al., 2023), LLaMATurk (Toraman, 2024), FinGPT (Luukkonen et al., 2023), and RedWhale (Vo et al., 2024) for Chinese, Turkish, Finnish, and Korean, respectively. These LLMs use one or more techniques such as tokenizer extension, secondary pretraining, and supervised fine-tuning. The key distinction of our work lies in its emphasis on developing bilingual LLMs, whereas the aforementioned efforts concentrate on creating monolingual LLMs.

Cahyawijaya et al. (2024) show that large language models can learn low-resource languages effectively using in-context learning and few-shot examples, improving performance through cross-lingual contexts without extensive tuning. Gurgurov et al. (2024) enhance multilingual LLMs for low-resource languages by using adapters with data from ConceptNet, boosting performance in sentiment analysis and named entity recognition.

[1] https://huggingface.co/nvidia/Nemotron-4-Mini-Hindi-4B-Base

[2] https://huggingface.co/nvidia/Nemotron-4-Mini-Hindi-4B-Instruct

[3] https://build.nvidia.com/nvidia/nemotron-4-mini-hindi-4b-instruct

[4] https://huggingface.co/Telugu-LLM-Labs/Indic-gemma-7b-finetuned-sft-Navarasa-2.0

[5] https://github.com/AI4Bharat/IndicInstruct

3 Methodology

In this section, we describe our methodology for adapting multilingual LLMs to target languages to improve performance in those languages. Specifically, we build a bilingual SLM that supports both Hindi and English. We conduct our adaptation experiments using the multilingual Nemotron-Mini-4B model (also known as Minitron-4B). The model undergoes continuous pre-training with an equal mixture of Hindi and English data, consisting of 200B tokens per language. The original Nemotron-4B model was primarily trained on English tokens and had seen only 20B Hindi tokens. Given the limited amount of Hindi data, adapting an existing multilingual model rather than training from scratch is an effective strategy, allowing us to leverage the knowledge learned from the pre-trained model. Additionally, as Nemotron-4B employs a large 256k tokenizer, we did not need to extend the tokenizer. The fertility ratio for Hindi text is 1.7, which is better than that of its Llama (2.64) and Gemma (1.98) counterparts.

3.1 Synthetic Data Curation

One of the key aspects of our work is the creation of a synthetic Hindi pre-training dataset. This synthetic data is generated using machine translation and transliteration. We first select high-quality English data sources and translate them into Hindi using a custom document translation pipeline. This pipeline preserves the document structure, including elements like bullet points and tables, and employs the IndicTrans2 model (Gala et al.) for sentence translation. However, since the translated data may contain noise, we use an n-gram language model to filter out low-quality samples. This model, trained on MuRIL-tokenized (Khanuja et al., 2021) real Hindi data, applies perplexity scores to identify and exclude noisy translations. Around 2% of the documents were discarded post-filtering.

The translated Hindi data comprises approximately 60 billion tokens. We then combine this synthetic data with around 40 billion real tokens (web-scraped data) to create a dataset totaling 100 billion Hindi tokens. Additionally, this entire Hindi text is transliterated into Roman script, expanding the total dataset to 220 billion tokens. The transliterated tokens are included to enable the model to support Hinglish queries. This Hindi data is further combined with 200 billion English tokens for continued pre-training. Including the English dataset helps prevent catastrophic forgetting of English capabilities and contributes to training stability. Fuzzy deduplication is performed on the entire text using NeMo-Curator[6] to eliminate similar documents. The real Hindi data sources include internal web-based datasets and Sangraha Corpus (Khan et al., 2024). The English dataset is a subset of the pre-training corpus used for the Nemotron-15B model. All the datasets used in this work are commercially friendly.

3.2 Continued Pre-training

The Nemotron-Mini-4B base model is used for continuous pre-training, and its architecture details are presented in Table 1. The Nemotron-Mini-4B model is derived from the Nemotron-15B model using compression techniques such as pruning and distillation, consisting of 2.6B trainable parameters (Muralidharan et al., 2024). Re-training is performed using a standard causal modeling objective. The dataset consists of 400B tokens, with an equal mix of Hindi and English. During batch sampling, greater weight is given to real data compared to synthetic data. We use the same optimizer settings and data split as (Parmar et al., 2024), with a cosine learning rate decay schedule from 2e-4 to 4.5e-7. This model is referred to as Nemotron-Mini-Hindi-4B, a base model where Hindi is the primary language. The re-training was performed using the Megatron-LM library (Shoeybi et al., 2020) and 128 Nvidia A100 GPUs.

3.3 Model Alignment

The first alignment stage is Supervised Fine-Tuning (SFT). We use a general SFT corpus with approximately 200k examples, comprising various tasks as outlined in (Adler et al., 2024). The model is trained for one epoch with a global batch size of 1024 and a learning rate in the range of [5e-6, 9e-7], using cosine annealing. Due to the lack of a high-quality Hindi SFT corpus, we leverage English-only data for SFT. We also experimented with translated English data (filtered using back-translation-based methods) for SFT, but did not observe any improvements with this addition. We found that using the English-only SFT corpus enhances instruction-following capabilities in Hindi, highlighting the cross-lingual transferability of these skills.

After SFT stage, the model undergoes a preference-tuning phase, where it learns from

[6] https://github.com/NVIDIA/NeMo-Curator

Base models	Metric	Nemotron-Mini-Hindi-4B	Nemotron-Mini-4B	Sarvam-1 2B	Gemma 2-2B	Openhathi	Llama-3.1 8B	Gemma 2-9B
IndicSentiment	F1 - NLU	84.31	72.47	96.36	91.90	72.89	92.06	94.90
IndicCopa	F1 - NLU	81.86	62.50	51.63	58.65	68.69	61.87	72.58
IndicXNLI	F1 - NLU	49.67	40.39	36.08	16.67	16.67	16.67	16.79
IndicXParaphrase	F1 - NLU	37.09	16.27	80.99	26.60	71.72	72.75	71.38
Indic QA (With Context) 1 shot	F1 - NLG	18.32	15.10	35.81	33.37	20.69	35.92	46.27
Indic Headline 1 shot	BLEURT - NLG	0.50	0.46	0.36	0.27	0.47	0.38	0.27
IndicWikiBio 1 shot	BLEURT - NLG	0.62	0.59	0.53	0.60	0.52	0.60	0.63
MMLU	Acc - NLU	49.89	38.20	45.65	35.05	32.27	44.84	55.08
BoolQ	Acc - NLU	71.71	70.79	56.08	66.00	58.56	61.00	61.00
ARC Easy	Acc - NLU	78.81	58.25	76.85	52.31	44.28	67.05	85.69
Arc Challenge	Acc - NLU	65.02	47.87	59.04	40.78	32.68	54.10	76.02
Hella Swag	Acc - NLU	31.66	25.31	37.13	27.50	25.59	33.50	42.40

Table 2: Performance metrics for various base models across different Hindi tasks. The results are zero-shot unless otherwise specified.

Instruct models	Metric	Nemotron-Mini-Hindi-4B	Nemotron-Mini-4B	Airavata	Navarasa 2B	Gemma-2 2B	Navarasa 7B	Llama-3.1 8B	Gemma-2 9B
IndicSentiment	F1 - NLU	97.62	90.01	95.81	93.62	94.32	95.99	98.59	99.09
IndicCopa	F1 - NLU	80.1	66.01	63.75	38.83	27.64	62.59	59.08	89.89
IndicXNLI	F1 - NLU	53.77	39.25	73.26	16.67	17.33	38.19	31.27	39.71
IndicXParaphrase	F1 - NLU	67.93	83.74	76.53	43.82	43.06	44.58	77.72	61.38
Indic QA (With Context) 1 shot	F1 - NLG	37.51	42.56	37.69	3.3	62.95	19.09	40.03	59.83
Indic Headline 1 shot	BLEURT - NLG	0.44	0.18	0.38	0.24	0.39	0.3	0.26	0.25
IndicWikiBio 1 shot	BLEURT - NLG	0.6	0.49	0.43	0.3	0.49	0.45	0.42	0.24
MMLU	Acc - NLU	50.5	38.66	34.96	23.1	39.39	40	45.85	57.35
BoolQ	Acc - NLU	67.86	60.00	64.5	60.31	70	78.1	80	84
ARC Easy	Acc - NLU	79.97	60.14	54	38.8	59.76	61.24	71.55	91.16
Arc Challenge	Acc - NLU	65.53	49.83	35.92	31.66	48.55	48.29	59.64	81.23
Hella Swag	Acc - NLU	39.9	39.69	25.37	25.3	34.7	30.8	35.5	54.6
IndicQuest (En)	Score (1-5)	4.01	3.94	3.75	3.78	4.1	4.07	4.2	4.4
IndicQuest (Hi)	Score (1-5)	4.15	2.72	3.1	3.18	3.58	3.6	4.02	4.23
SubjectiveEval (Hi)	Score (1-5)	4.35	1.64	2.24	1.75	3.66	2.97	3.98	4.5

Table 3: Performance metrics for various instruct models across different Hindi tasks. The results are zero-shot unless otherwise specified.

Task	Nemotron-Mini-Hindi-4B-Base	Nemotron-Mini-4B-Base	Gemma-2 2b
MMLU (5)	56.37	58.60	51.3
arc_challenge (25)	46.08	50.90	55.4
hellaswag (10)	74.64	75.00	73
truthfulqa_mc2 (0)	41.05	42.72	-
winogrande (5)	70.09	74.00	70.9
xlsum_english (3)	29.71	29.62	-

Table 4: Performance of base models on English Benchmarks

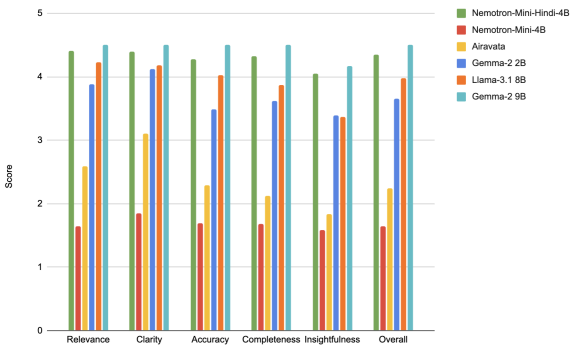

Figure 2: Comparison of different instruct models on various parameters using SubjectiveEval.

triplets consisting of a prompt, a preferred response, and a rejected response. In this stage, we apply the Direct Preference Optimization (DPO) (Rafailov et al., 2024) algorithm, which trains the policy network to maximize the reward difference between the preferred and rejected responses. We train the model for one epoch with a global batch size of 512 and a learning rate in the range of [9e-6, 9e-7], utilizing cosine annealing. For the DPO stage, we use approximately 200k English samples and 60k synthetic Hindi samples. The synthetic Hindi samples were created by translating the English samples and then filtered using back-translation methods. We observe that incorporating synthetic Hindi samples during this stage improves the overall performance of the model. The aligned model is referred to as Nemotron-Mini-Hindi-4B-Instruct. Both the SFT and DPO stages are carried out using Nemo Aligner (Shen et al., 2024) and 64 Nvidia A100 GPUs.

3.4 Evaluation Datasets

We evaluate Nemotron-Mini-Hindi-4B and other multilingual LLMs using both native Hindi benchmarks and translated English benchmarks. The native benchmarks include tasks from IndicXTREME, IndicNLG, and IndicQuest, while the translated English benchmarks include popular datasets like MMLU and Hellaswag. Additionally, we curate an open-ended QnA dataset termed SubjectiveEval to assess the model's generation capabilities in the Hindi language. Human evaluation is also conducted using the translated MT-Bench dataset.

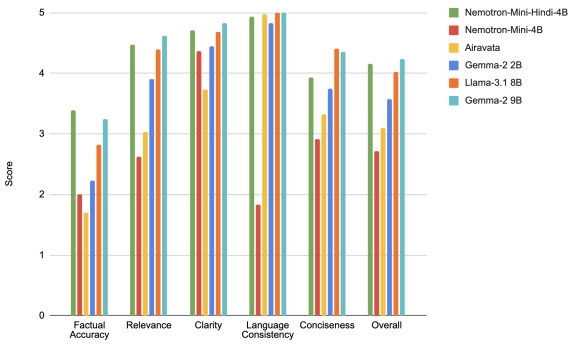

Figure 3: Comparison of different instruct models on various parameters using IndicQuest-Hi.

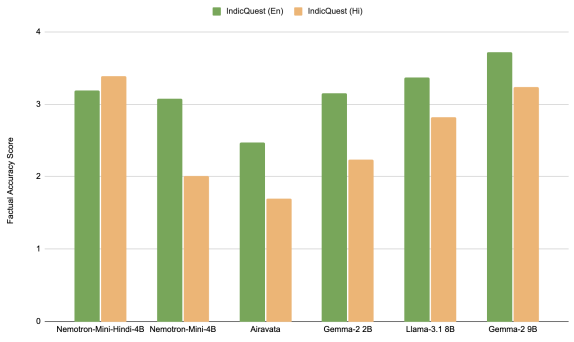

Figure 4: Comparison of different instruct models on Factuality score of IndicQuest. The ground truth answers from IndicQuest are provided as a reference to GPT4 for better scoring. The Nemotron-Mini-Hindi-4B provides comparable scores for Hindi and English whereas other models provide better factuality for English.

- **IndicXTREME**: The benchmark consists of different Natural Language Understanding (NLU) tasks in Indic languages (Doddapaneni et al., 2023). We consider different tasks like IndicSentiment, IndicCopa, IndicXNLI, and IndicXParaphrase.

- **IndicNLG**: The IndicNLG benchmark (Kumar et al., 2022) consists of various tasks for evaluating the generation capabilities of the model. We consider IndicHeadline, IndicWikiBio, and IndicQA covering text summarization and question-answering tasks.

- **IndicQuest**: IndicQuest (Rohera et al., 2024) is a gold-standard fact-based question-answering benchmark designed to evaluate multilingual language models ability to capture regional knowledge across various Indic languages. It focuses on factual questions related to India in domains such as Literature, History, Geography, Politics, and Economics. The dataset is available in English as well as several Indic languages, including Hindi, allowing for language-specific evaluations. For LLM-as-a-judge evaluation, the ground truth facts are passed to the evaluator LLM as a reference.

- **SubjectiveEval**: This in-house Hindi evaluation dataset features open-ended questions across various Indian domains, including History, Geography, Agriculture, Food, Culture, Religion, Science and Technology, Mathematics, and Thinking Ability. It offers broader coverage compared to the fact-based questions in IndicQuest. It assesses a model's understanding, generative capabilities, coherence, and insightfulness. Questions include 'what', 'how', and 'why' types, varying from brief one-word answers to detailed explanations. The dataset also tests analytical and problem-solving skills with hypothetical scenarios. Model responses are evaluated using an LLM as a judge.

- **Translated English Benchmarks**: We use translated versions of popular benchmarks for exhaustive evaluation of our models. The benchmarks include MMLU, Hella Swag, BoolQ, Arc-Easy, and Arc-Challenge.

- **Human Evaluation**: For human evaluation, we utilized a translated version of the multi-turn MT-Bench dataset (Zheng et al., 2023). The prompts were first translated into Hindi using the Google Translate API and then manually filtered to remove problematic prompts or those relying on English-specific semantics. During evaluation, human judges conducted A/B testing, where they were presented with randomized, pair-wise model responses for comparison.

4 Results and Discussion

The results for the base models are shown in Table 2 and Table 4. The Nemotron-Mini-Hindi-4B Base delivers state-of-the-art performance on nearly all benchmarks compared to similarly sized models. Additionally, it outperforms larger models like Gemma-2-9B and Llama-3.1-8B on more than half of the benchmarks. Hindi-specific continued pre-training significantly enhances the model's performance on Hindi tasks compared to the base

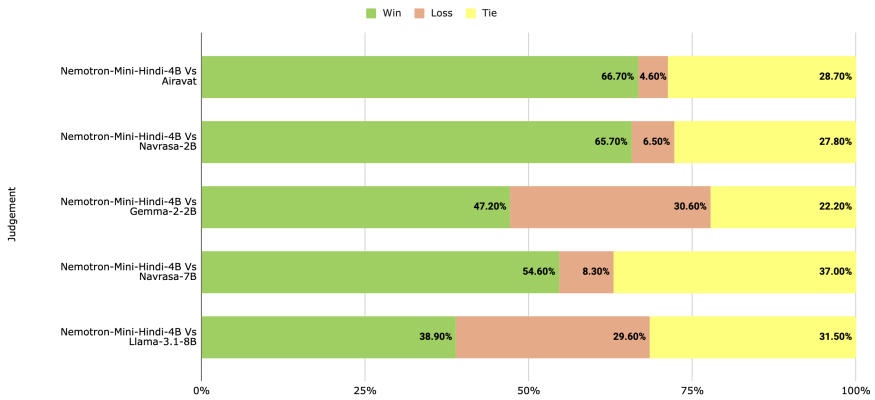

Figure 5: Results of human evaluation on translated MT-Bench. A win indicates Nemotron-Mini-Hindi-4B model is preferred.

Nemotron-Mini-4B model. There is some degradation on English benchmarks, though the results remain competitive. This underscores the importance of dual-language continued pre-training.

We observe similar results with the instruct model on IndicXTREME, IndicNLG, and translated English benchmarks. The results are presented in Table 3. The instruct model is also evaluated using LLM-as-a-judge on IndicQuest and SubjectiveEval. On these benchmarks, we see improvements in both English and Hindi compared to the Nemotron-Mini-4B-Instruct model. The model outperforms all baseline models except for Gemma-2-9B. Notably, we observe improvements in the model's factuality and language consistency. These results are shown in Figure 2, 3, and 4. Furthermore, during human evaluations, responses from Nemotron-Mini-4B-Hindi were consistently preferred over those from other models, as shown in Figure 5.

5 Conclusion

We present Nemotron-Mini-Hindi-4B-Base and Nemotron-Mini-Hindi-4B-Instruct, state-of-the-art SLMs primarily designed for the Hindi language. These models have been continuously pre-trained and aligned using a combination of Hindi and English data. The Hindi corpus includes both real and synthetic data, with the synthetic data generated through translation. The models outperform similarly sized models on various Hindi benchmarks, as assessed through reference-based and LLM-as-a-judge evaluations. They also perform competitively on English benchmarks. We emphasize the importance of pre-training to reduce hallucinations and enhance the factuality of the models.

Limitations

The model was trained on internet data that includes toxic language and biases, which means it might reproduce these biases and generate toxic responses, particularly if prompted with harmful content. It may also produce inaccurate, incomplete, or irrelevant information, potentially leading to socially undesirable outputs. The problem could be worsened if the suggested prompt template is not used.

To mitigate these issues to some extent, we have implemented safety alignment during the DPO stage to guide the model away from responding to toxic or harmful content. Additionally, we conduct safety evaluations using benchmarks such as Aegis[7] (Ghosh et al., 2024), Garak[8] (Derczynski et al., 2024), and Human Content red-teaming, and our findings indicate that the model's responses remain within permissible limits.

Acknowledgements

This work would not have been possible without contributions from many people at NVIDIA. To mention a few: Asif Ahamed, Ayush Dattagupta, Umair Ahmed, Yoshi Suhara, Ameya Mahabaleshwarkar, Zijia Chen, Varun Singh, Vibhu Jawa, Saurav Muralidharan, Sharath Turuvekere Sreenivas, Marcin Chochowski, Rohit Watve, Oluwatobi Olabiyi, Mostofa Patwary, and Oleksii Kuchaiev.

[7] https://huggingface.co/datasets/nvidia/Aegis-AI-Content-Safety-Dataset-1.0

[8] https://github.com/leondz/garak

References

Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, Bryan Catanzaro, Sharon Clay, Jonathan Cohen, et al. 2024. Nemotron-4 340b technical report. *arXiv preprint arXiv:2406.11704*.

Abhinand Balachandran. 2023. Tamil-llama: A new tamil language model based on llama 2. *arXiv preprint arXiv:2311.05845*.

Samuel Cahyawijaya, Holy Lovenia, and Pascale Fung. 2024. Llms are few-shot in-context low-resource language learners. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 405–433.

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

Leon Derczynski, Erick Galinkin, Jeffrey Martin, Subho Majumdar, and Nanna Inie. 2024. garak: A framework for security probing large language models. *arXiv preprint arXiv:2406.11036*.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards leaving no indic language behind: Building monolingual corpora, benchmark and models for indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jay Gala, Pranjal A Chitale, AK Raghavan, Varun Gumma, Sumanth Doddapaneni, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, et al. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*.

Jay Gala, Thanmay Jayakumar, Jaavid Aktar Husain, Mohammed Safi Ur Rahman Khan, Diptesh Kanojia, Ratish Puduppully, Mitesh M Khapra, Raj Dabre, Rudra Murthy, Anoop Kunchukuttan, et al. 2024. Airavata: Introducing hindi instruction-tuned llm. *arXiv preprint arXiv:2401.15006*.

Shaona Ghosh, Prasoon Varshney, Erick Galinkin, and Christopher Parisien. 2024. Aegis: Online adaptive ai content safety moderation with ensemble of llm experts. *arXiv preprint arXiv:2404.05993*.

Daniil Gurgurov, Mareike Hartmann, and Simon Ostermann. 2024. Adapting multilingual llms to low-resource languages with knowledge graphs via adapters. In *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, pages 63–74.

Raviraj Joshi. 2022. L3cube-mahanlp: Marathi natural language processing datasets, models, and library. *arXiv preprint arXiv:2205.14728*.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, Mitesh M Khapra, et al. 2024. Indicllmsuite: A blueprint for creating pre-training and fine-tuning datasets for indian languages. *arXiv preprint arXiv:2403.06350*.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Aman Kumar, Himani Shrotriya, Prachi Sahu, Amogh Mishra, Raj Dabre, Ratish Puduppully, Anoop Kunchukuttan, Mitesh M Khapra, and Pratyush Kumar. 2022. Indicnlg benchmark: Multilingual datasets for diverse nlg tasks in indic languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5363–5394.

Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, et al. 2023. Fingpt: Large generative models for a small language. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726.

Nick Mecklenburg, Yiyou Lin, Xiaoxiao Li, Daniel Holstein, Leonardo Nunes, Sara Malvar, Bruno Silva, Ranveer Chandra, Vijay Aski, Pavan Kumar Reddy Yannam, et al. 2024. Injecting new knowledge into large language models via supervised fine-tuning. *arXiv preprint arXiv:2404.00213*.

Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. 2024. Compact language models via pruning and knowledge distillation. *arXiv preprint arXiv:2407.14679*.

Jupinder Parmar, Shrimai Prabhumoye, Joseph Jennings, Mostofa Patwary, Sandeep Subramanian, Dan Su, Chen Zhu, Deepak Narayanan, Aastha Jhunjhunwala, Ayush Dattagupta, et al. 2024. Nemotron-4 15b technical report. *arXiv preprint arXiv:2402.16819*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Pritika Rohera, Chaitrali Ginimav, Akanksha Salunke, Gayatri Sawant, and Raviraj Joshi. 2024. L3cube-indicquest: A benchmark question answering dataset for evaluating knowledge of llms in indic context. *arXiv preprint arXiv:2409.08706*.

Gerald Shen, Zhilin Wang, Olivier Delalleau, Jiaqi Zeng, Yi Dong, Daniel Egert, Shengyang Sun, Jimmy Zhang, Sahil Jain, Ali Taghibakhshi, et al. 2024. Nemo-aligner: Scalable toolkit for efficient model alignment. *arXiv preprint arXiv:2405.01481*.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. Megatron-lm: Training multi-billion parameter language models using model parallelism. *Preprint*, arXiv:1909.08053.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Cagri Toraman. 2024. Llamaturk: Adapting open-source generative large language models for low-resource language. *arXiv preprint arXiv:2405.07745*.

Anh-Dung Vo, Minseong Jung, Wonbeen Lee, and Daewoo Choi. 2024. Redwhale: An adapted korean llm through efficient continual pretraining. *arXiv preprint arXiv:2408.11294*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.