PARAMANU: Compact and Competitive Monolingual Language Models for Low-Resource Morphologically Rich Indian Languages

Mitodru Niyogi[1] **Eric Gaussier**[1] **Arnab Bhattacharya**[2]
[1]Université Grenoble Alpes, CNRS, Grenoble INP, LIG, 38000 Grenoble, France
[2]Dept. of Computer Science and Engineering, Indian Institute of Technology Kanpur, India
mitodru.niyogi@cnrs.fr, eric.gaussier@imag.fr, arnabb@iitk.ac.in

Abstract

Multilingual large language models (LLMs) are expensive to pretrain and often suffer from imbalances across languages and datasets, English-centric bias, tokenizer oversegmentation for morphologically rich low-resource languages, and the curse of multilinguality. We introduce PARAMANU, a family of Indian language-only autoregressive language models trained from scratch on open-source language-specific data for the five most spoken Indian languages: Bangla (Bengali), Hindi, Marathi, Tamil, and Telugu. All models are designed for *affordability* and are trained on a *single GPU* with a budget *under $1,000*, allowing under-resourced researchers to build competitive language models. To address low-resource challenges, we develop morphology-aligned, low-fertility *tokenizers*, and propose an interpolation-based method for token position indices in RoPE based scaling to train longer sequences efficiently. We also create instruction-tuning datasets in Bangla that are then translated to the other four languages. Despite their small size (108M-367M parameters), *Paramanu* achieves a strong performance-efficiency tradeoff and outperforms most larger multilingual models up to 8B across all five languages. The models and datasets are available at: https://huggingface.co/collections/mitodru/paramanu.

1 Introduction

Despite the existence of over 7,000 languages globally, current NLP and GenAI technologies remain heavily skewed towards English and other high-resource European languages, leaving a significant portion of the world's population, particularly speakers of global south languages, underserved (Schwartz, 2022; Nekoto et al., 2020; Choudhury, 2023). Indian languages, spoken by approximately 1.4 billion people, are among the most neglected, despite the fact that Hindi and Bangla (Bengali) are respectively the 5th and 6th most spoken[1] languages globally. Challenges such as lack of high-quality datasets, poor tokenization, and limited representation in pretraining corpora render Indian languages "low-resource" (Tsvetkov, 2017; Singh, 2008); being morphologically rich further impedes their performance (Joshi et al., 2020; Goyal et al., 2022; Nigatu et al., 2024).

Large language models (LLMs) like GPT-2 (Radford et al., 2019), LLaMa (Touvron et al., 2023), GPT-NeoX (Black et al., 2022), OPT (Zhang et al., 2022), Falcon (Almazrouei et al., 2023), and PaLM (Chowdhery et al., 2023) are predominantly trained on English and Latin-script languages, showing significantly degraded performance on Indian and other low-resource languages (Bang et al., 2023; Lai et al., 2023a). This disparity persists even in multilingual decoder-only LLMs (e.g., Bloom (Workshop et al., 2023), xGLM (Lin et al., 2022), mGPT (Shliazhko et al., 2024), Aya23 (Aryabumi et al., 2024), Llama-3 (Grattafiori et al., 2024), Llama-3.2 (Meta AI, 2024)). These models have English-centric bias, and "think in English" (Schut et al., 2025; Guo et al., 2024a), which often causes them to perform worse in non-Latin script languages (Shafayat et al., 2024; Shi et al., 2023; Huang et al., 2023; Bang et al., 2023). As they also suffer from data and language-dependent imbalance (Dangarikar et al., 2024), a fits-all-language tokenizer often leads to bias, over-segmentation (Ahuja et al., 2023), unfair representation (Pfeiffer et al., 2021), language confusion (Marchisio et al., 2024) and reduced fluency (Guo et al., 2024b). This results in high token fertility for Indian languages, and increased inference and training costs.

Adapting existing LLMs to Indian languages through continual pretraining (Zheng et al., 2024a; Ji et al., 2025; Alves et al., 2024) and fine-tuning

[1]https://www.babbel.com/en/magazine/the-10-most-spoken-languages-in-the-world

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 41431–41458

(Lu et al., 2024; Zheng et al., 2024b) presents multiple challenges, including requirement of large data and compute, vocabulary extension, embedding alignment, and risk of catastrophic forgetting (Zheng et al., 2024a; Ahuja et al., 2023). Moreover, this adaptation assumes the suitability of English-centric foundations, which may not generalize well to Indian typologies and scripts.

We introduce here PARAMANU[2], the first family of Indian language-only, openly licensed, non-commercial (CC BY-NC-SA 4.0) sub-400M decoder language models trained from scratch on fully open-source, language-specific Indian data. *Paramanu* comprises monolingual generative models for the five most spoken Indian languages: Bangla (Bengali), Hindi, Marathi, Tamil, and Telugu ranging from 108M to 367M parameters, as well as a multilingual model covering Assamese, Bangla, Sanskrit, Konkani, Maithili and Odia. We use low-fertility, language-specific tokenizers to reduce training and inference cost and latency. This approach avoids cross-language data imbalance, allows effective preprocessing of smaller corpora (including low-resource settings), and provides a controlled foundation for analyzing LLM behavior and developing downstream adaptations. We show that language-specific models achieve strong performance even under severe constraints[3] for low-resource, morphologically rich languages. Across a comprehensive evaluation against monolingual and multilingual models up to 8B parameters, *Paramanu* achieves an efficient performance-cost tradeoff, outperforming most larger models and, to our knowledge, all LLMs under 3B parameters.

We summarize our contributions as follows:

1. We developed *Paramanu*, the first from-scratch, Indian language-only open-source sub-400M decoder LMs for five most spoken Indian languages, thereby empirically demonstrating that small monolingual models can outperform larger multilingual models, making them broadly usable by NLP researchers working on Indian languages.
2. We developed morphology-aligned, low-fertility Indian tokenizers by combining Unigram and BPE tokens.
3. We developed an interpolation method, inspired from Chen et al. (2023), for the position indices of tokens in RoPE based scaling for training longer sequences on a single GPU.
4. We cleaned the training corpus and developed novel instruction-tuning datasets in Bangla, which were then machine translated and used for Hindi, Marathi, Tamil, and Telugu to further align our models with human instructions. Our datasets are available at https://huggingface.co/collections/mitodru/paramanu.

The remainder of the paper is organized as follows: Section 2 discusses prior work; Section 3 details data, tokenization, and model design; Section 4 presents experiments, training, evaluations, and ablations; Section 5 discusses the results; and Section 6 concludes.

2 Related Work

Multilingual large language models (LLMs) such as Bloom (Workshop et al., 2023), xGLM (Lin et al., 2022), and Sarvam 2B (Sarvam2B, 2024) have made significant progress in scaling decoder-only models across multiple languages. However, many of these models remain heavily English-biased: for instance, Llama-3.2 (Dubey et al., 2024) includes only 8% non-English tokens, while Sarvam 2B uses 40-50% English data. Such imbalances, together with tokenizer over-segmentation, disproportionately affect low-resource, morphologically rich languages. Efforts to adapt English-centric models for Indian languages include Airavata (Gala et al., 2024), OpenHathi (SarvamAI, 2023), TigerLLM (Raihan and Zampieri, 2025), and Nemotron-Hindi (Joshi et al., 2025), which extend vocabularies and leverage fine-tuning techniques like LoRA (Hu et al., 2022) and QLoRA (Dettmers et al., 2023). Dedicated Indian-language LLMs trained from scratchsuch as Aya23 8B (Aryabumi et al., 2024), mGPT (Shliazhko et al., 2024), and Sarvam 2B (Sarvam2B, 2024) still rely heavily on English data and struggle to generate high-quality text in Indian languages.

Massively multilingual models (MMTs) (Devlin et al., 2019; Conneau et al., 2020a; Xue et al., 2021) are pretrained on large corpora across many languages but often lack alignment between distant languages, resulting in poor transfer performance (Lauscher et al., 2020). Studies attribute this to lower-quality tokenization per language (Rust et al., 2021) and show that adding multilingual data helps low-resource languages only until

[2]Available at: https://huggingface.co/collections/mitodru/paramanu.

[3]Typically, a single GPU and <$1,000 budget for training.

model capacity is reached, while consistently degrading performance for high-resource languages (Chang et al., 2024). Indic NLP research also suffers from a lack of culturally and linguistically relevant datasets (Doddapaneni et al., 2023; Khan et al., 2024), as most supervised datasets are translations from English. Recent efforts have explored building monolingual autoregressive LMs from scratch, such as German LLäMmlein (Pfister et al., 2025), BanglaT5 (Bhattacharjee et al., 2023), and BanglaByT5 (Bhattacharyya and Bhattacharya, 2025). Unlike BanglaByT5, which compared only models below 1B parameters, our work evaluates *Paramanu* against models up to 8B parameters, demonstrating the effectiveness of small, language-specific LLMs under resource constraints. This family of models can be used and extended by any NLP group working on Indian languages.

3 Methodology

In this section, we discuss details regarding our datasets, preprocessing, novel tokenization, multilingual instruction dataset creation, and context scaling with tokens positional interpolation.

3.1 Dataset for Pretraining

The pretrained data of 54.6 GB UTF-8 bytes for the 5 Indian languages (Bangla, Hindi, Marathi, Tamil, Telugu) was split into 95% training and 5% validation to retain as much data as possible, since the goal is to take a step toward developing effective pretrained generative language models from scratch for the 5 most spoken Indian languages. The pretraining corpus consists of web-scraped news, blogs, and Wikipedia from IndicCorp v2 (Doddapaneni et al., 2023) for Marathi, Tamil, and Telugu, which was used to train IndicBERT-v2, and Bangla literature from Vacaspati (Bhattacharyya et al., 2023), used in training Bangla Electra. IndicCorp v2 also includes Indian language data from Wikipedia and OSCAR (Suárez et al., 2019). Dataset details are in Table 1.

3.2 Data Cleaning

Data curation and cleaning are important for low resource languages to improve the signal/noise ratio (Kreutzer et al., 2022). Following prior work (Doddapaneni et al., 2023; Abadji et al., 2022), we perform regex-based filtering of HTML/XML tags, emails, links, emojis, personal info, and remove non-literal and foreign-script characters. Language identification is done using `cld3`[4] and IndianLID-FTN (Madhani et al., 2023) to discard non-target languages. We filter toxic content using Team et al. (2022), normalize whitespace and Unicode, and deduplicate paragraphs using 128-bit MurmurHash[5]. For Indian scripts (Bangla, Devanagari, Tamil, and Telugu), sentence splitting uses language-specific punctuation (danda "।" for Bangla and Devanagari scripts).

3.3 Tokenization

To improve morphological subword representations for Indian languages, we employ a hybrid tokenization approach that fuses vocabularies from independently trained SentencePiece (Kudo and Richardson, 2018) models using Byte Pair Encoding (BPE, Sennrich et al., 2016) and the Unigram Language Model (Unigram LM, Kudo, 2018). This design is motivated by Bostrom and Durrett (2020), who show that Unigram LM produces subword units that better capture morphological structure through global optimization and probabilistic pruning, yielding cleaner subword inventories than greedy merge-based methods.

Both tokenizers are converted to SentencePieces `ModelProto` format, which serializes the vocabulary, subword scores, normalization rules, and special tokens, and the BPE vocabulary is augmented with all Unigram LM tokens not already present. The added Unigram LM tokens are assigned their original Unigram LM scores in SentencePiece, which are computed as log-probabilities of empirical frequencies:

$$\text{score}_u(t) = \log f_t$$

where $f_t \in (0, 1)$ is the frequency (*i.e.*, the normalized number of occurrences) of token t. SentencePiece furthermore relies on a Viterbi-style algorithm for tokenization, using, for BPE scores, the opposite of the rank $(-r_t)$ of each token in a frequency-sorted (descending) list. By Zipf's law: $r_t = \frac{1}{H_N f_t}$, where N is the number of tokens considered and $H_N = \sum_{k=1}^{N} \frac{1}{k}$ is the N^{th} harmonic number. The function

$$g(f_t) = \log f_t - (-\frac{1}{H_N f_t}) = \log f_t + \frac{1}{H_N f_t}$$

is decreasing from $+\infty$ to $(1 + \log H_N^{-1})$ on $(0, H_N^{-1}]$, and increasing from $(1 + \log H_N^{-1})$ to

[4] https://github.com/google/cld3
[5] https://pypi.org/project/mmh3/

Language	Family	Script	Corpus Source	Corpus Size (GB)	#Sentences	#Speakers
Bangla	Indo-European	Bangla	Vacasapati + Wikipedia	3.6	22,533,608	300 M
Hindi	Indo-European	Devanagari	IITB monolingual	15.8	52,124,643	692 M
Marathi	Indo-European	Devanagari	Indian Corp v2	12.5	34,567,839	99 M
Tamil	Dravidian	Tamil	Indian Corp v2	10.7	27,872,768	77 M
Telugu	Dravidian	Telugu	Indian Corp v2	13.5	40,241,847	95 M

Table 1: Pretraining data details after data cleaning along with language families, scripts, and speaker estimates. Speaker data is from the Indian Census 2011.

H_N^{-1} on $[H_N^{-1}, 1)$. Furthermore, as $1 + \log H_N^{-1} > 0$ for $N \geq 8$, there exist $K_1 \in (0, H_N^{-1})$ and $K_2 \in (H_N^{-1}, 1)$ such that $g(K_1) = g(K_2) = 0$. This shows that Unigram LM tokens are privileged over BPE tokens in $(0, K_1) \cup (K_2, 1)$, and that BPE tokens are privileged over Unigram LM tokens in (K_1, K_2). As Unigram LM and BPE share the most frequent tokens, the Unigram LM tokens added to the BPE tokens correspond to less frequent, longer tokens. Thus, in practice, the segmentation for most words mostly relies on BPE in the first merges, and then on Unigram LM, if possible, for later merges where morphologically meaningful substrings are captured (Bostrom and Durrett, 2020). For many words, the resulting segmentation is identical to BPE alone (rows 4-5 in Table 6 in Appendix A), but for substrings where Unigram LM tokens better match common morphemes, the hybrid tokenizer selects these as atomic units, producing more morphologically coherent subwords.

Table 6 in Appendix A illustrates the tokenization behavior of our hybrid tokenizer compared to standard BPE across several Indian languages. In Telugu, the word ఆధారపడతాము (ādhārapaḍatāmu) is segmented by BPE as ['ఆధార', 'ప', 'డ', 'తాము'] (['ādhāra', 'pa', 'ḍa', 'tāmu']), which breaks the root ఆధారపడ ('ādhārapaḍa') into separate tokens. In contrast, our hybrid tokenizer produces ['ఆధారపడ', 'తాము'] (['ādhārapaḍa', 'tāmu']), preserving the root as a single unit and the suffix separately, maintaining the morphological and semantic structure of the word. For Tamil, the word பயணித்தார்கள் (payaṇittārkaḷ) is segmented by BPE as ['பயண', 'இத்தார்கள்'] (['payaṇa', 'ittārkaḷ']), splitting the verb root பயணித்த (payaṇitta) and the past-tense participle plus plural suffix into unnatural fragments. The hybrid tokenizer segments it as ['பயணித்த', 'ஆர்கள்'] (['payaṇitta', 'ārkaḷ']), keeping the root plus tense marker பயணித்த together and the plural marker ஆர்கள் as a separate token, which better reflects the underlying morphological units. Across these examples, the hybrid tokenizer consistently preserves stems and frequent suffixes as atomic subwords, whereas BPE often produces over-fragmented tokens. By combining BPE with Unigram LM tokens, it increases lexical coverage by representing both frequent and rare morphemes as reusable units. This enables more compact token sequences, reduces embedding redundancy, and generates subword representations that better align with the semantic and morphological structure of morphologically rich Indian languages.

During pre-tokenization, we apply NFC normalization, digit splitting, and byte fallback for unknown UTF-8 characters. Our tokenizers achieve the least fertility scores across all five languages compared to Sarvam 2B (Sarvam2B, 2024), Llama-3.1 (Dubey et al., 2024), Gemma-2 (Team et al., 2024), and GPT-4o (shown in Fig. 1).

3.4 Instruction Tuning Datasets

We constructed 23K instructions for Bangla from three sources: 5K human-authored instructions (on culture, literary, practical domain) by 20 native Bangla-speaking annotators, following guidelines detailed in Appendix B, 15K translated instructions from Dolly (Conover et al., 2023), and 3K self-instruct-generated samples (Wang et al., 2023). These were translated to Hindi, Marathi, Tamil, and Telugu using Google Translate[6] with manual post-editing[7] The details of this dataset are given in Table 7 in the Appendix. We acknowledge that using Google Translate for Hindi, Marathi, Tamil, and Telugu may introduce linguistic artifacts. However, translation-based supervision is a widely adopted strategy in multilingual and low-resource NLP (Conneau et al., 2020b; Xue et al., 2021; Chung et al., 2022), particularly when high-quality native instruction data is un-

[6] https://cloud.google.com/

[7] Available at: https://huggingface.co/collections/mitodru/paramanu.

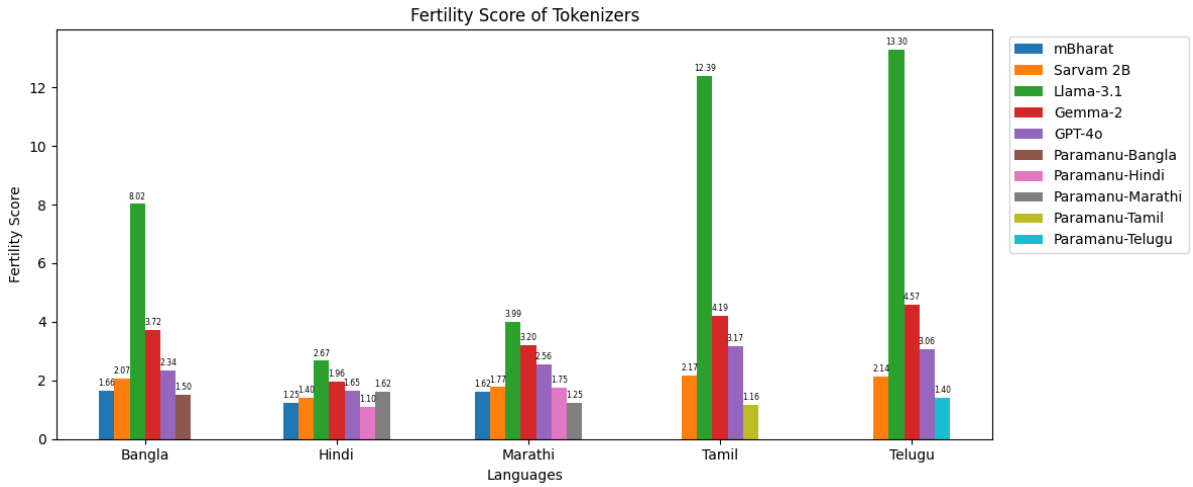


Figure 1: Fertility score of our tokenizer versus LLMs across languages of 4 scripts (Bangla, Devanagari, Tamil, and Telugu). LLMs score are reported from (Sarvam2B, 2024).

available. To mitigate this, we applied manual post-editing and evaluation was conducted on independently curated target-language benchmarks. Given the scarcity of high-quality instruction-following datasets, translation provides a practical initialization strategy rather than a replacement for culturally grounded native data.

3.5 Context Scaling with RoPE Embeddings for Efficient Pretraining

We employ a scaled variant of Rotary Positional Embeddings (RoPE; Su et al., 2022) with a base value of $\theta = 10{,}000$. Inspired from Chen et al. (2023) to support pretraining with large context lengths on hardware-constrained settings (e.g., a single A100 40GB GPU), we introduce a *shrinking factor* that scales the input token position ids before the RoPE methodology is applied. This *shrinking factor* is defined as the ratio of the target context length y to a fixed *permissible_context_size_length* L, which corresponds to the maximum context length that the available hardware can accommodate. All other training hyperparameters such as batch size, vocabulary size, and model dimensions remain unchanged. Formally, for each token position p, we compute a scaled position p' as:

$$p' = \frac{p}{\alpha} = \frac{p \cdot L}{y}$$

For example, with a target context length of $y = 4096$ and a permissible length of $L = 256$, the shrinking factor is $\alpha = \frac{4096}{256} = 16$. A token at position $p = 4000$ is mapped to $p' = \frac{4000}{16} = 250.00$, and its neighbor at $p = 4001$ maps to $p' \approx 250.06$. This ensures all scaled positions lie within the permissible range $[0, L]$. Fig. 2 illustrates the process.

Note that, as positional embeddings can be applied to non-integer positions using RoPE, the self-attention score is only dependent on the relative position of the tokens through trigonometric functions. This independence on y is crucial for stability of the attention scores and contrasts with the extrapolation approach of Su et al. (2022). As such, the positional interpolation of Chen et al. (2023) is more likely to behave well in practice, even though stability may not be guaranteed if, for example, the dimension of the attention head is large or if the q and k vectors have large coordinates.

4 Experiments

Our monolingual PARAMANU models are based on transformer (Vaswani et al., 2017) causal decoder architecture (Radford et al., 2019). Following Chinchilla (Hoffmann et al., 2022a) and LLaMa (Touvron et al., 2023), we used RMSNorm (Zhang and Sennrich, 2019) with norm_epsilon =1e-5, SwiGLU (Shazeer, 2020) activation function, and an activation hidden size of $\sim \frac{8}{3}d$. Following (Chowdhery et al., 2023), we removed all biases from dense layers to improve the training stability. We also used weight tying (Press and Wolf, 2017) to improve the performance of language models by sharing the weights of the embedding and softmax layers. The model parameter configuration is shown in Table 15 in Appendix D.

4.1 Training

We pretrained our models using the AdamW optimizer (Loshchilov and Hutter, 2019), with β_1 =

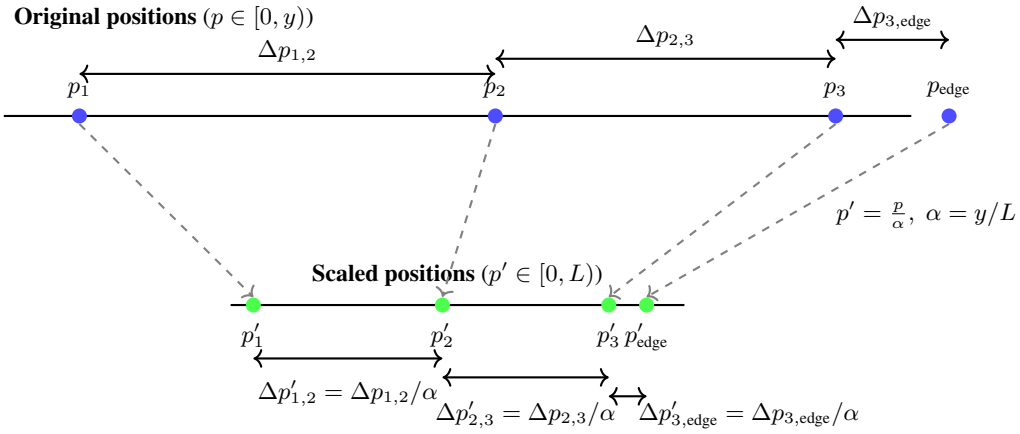


Figure 2: RoPE context scaling via positional interpolation. Tokens in the original context (blue) are linearly mapped to a hardware-supported range (green). Absolute distances shrink, relative offsets $\Delta p' = \Delta p/\alpha$ are preserved. Brackets for $\Delta p'$ are spaced to avoid overlap with token labels.

0.9, β_2 = 0.95, eps = 10^{-5}. We use a cosine learning rate schedule, with warmup of 1000 steps, and decay final learning rate down to 10% of the peak learning rate. We use a weight decay of 0.1 and gradient clipping of 1.0. To further speedup training, we also used BF16 mixed precision training. We performed hyperparameter tuning on 15M models and used the concept of μP transfer (Yang et al., 2021) to transfer the learned hyperparameters to our bigger models. All models are pretrained for 100K training steps except Hindi 367M (150K). For instruction-tuning, we followed Taori et al. (2023). Further details are in Appendix D.

4.2 Evaluation

We evaluate our models on perplexity and downstream tasks including QA, NLI, and commonsense reasoning, across five Indian languages. We also performed human evaluation for Bangla and Hindi as discussed in Appendix C.3. Comparisons are made with 25 multilingual and Indian-adapted LLMs (200M–8B params) such as BanglaT5, BanglaByT5, Bloomz, xGLM, LLaMA-3/3.2, mGPT, Sarvam 2B, Aya23, and fine-tuned models such as OpenHathi, Nemotron-Hindi (Joshi et al., 2025), Airavata (Gala et al., 2024), Tiger-LLM, and Indic-Gemma-Navrasa (Telugu-LLM-Labs, 2025). Models are grouped by size.

Benchmarks. We use MMLU (Hendrycks et al., 2021), ARC (Clark et al., 2018), and Belebele (Bandarkar et al., 2024) for all languages, and language-specific datasets: HellaSwag (Hindi), XNLI (Conneau et al., 2018), XStoryCloze (Lin et al., 2022) (Hindi, Telugu), and XCOPA (Ponti et al., 2020) (Tamil) as benchmarks. Evaluation uses machine translated datasets from Lai et al. (2023b) and LM Evaluation Harness (LMEval) (Sutawika et al., 2024).

4.3 Results

Table 2 summarizes average performance across benchmark tasks in five Indian languages (detailed results are given in Appendix Tables 8, 10, 9, 11, 12). The instruction-tuned Paramanu models outperform, on each language, the 15 LLMs with less than 3B parameters, with the exception of Sarvam 2B which slightly outperforms the Tamil and Telugu Paramanu models; it is however trained on 4T tokens, 50× more than the Paramanu models. The Paramanu models furthermore outperform the larger models OpenHathi-7B and Airavata-7B adapted on Marathi and Hindi through continual pretraining, as well as 6 out of 10 larger (above 3B parameters) language models, including LLMs using up-sampled low-resource data as xGLM which is trained with 30 languages. Although Llama-3-8B and Bloomz-7B lead overall due to large-scale instruction tuning on xP3, our models remain competitive with all other models. Our reliance on translated benchmarks (e.g., MMLU, ARC, HellaSwag) may favor multilingual models trained on predominantly Western data, potentially providing them an advantage on culturally Western content. Thus, our evaluation may represent an upper bound for such baselines and likely penalizes

Model	Size	#langs	Bangla	Devanagari		Tamil	Telugu	Training	Context	Pretrained Tokens	
				Marathi	Hindi			Hours	Size	Total	Indian
Paramanu-Bangla	108M	1	25.22	-	-	-	-	42.75	1024	26.21B	26.21B
Paramanu-Bangla-instruct	108M	1	**29.52**	-	-	-	-	+0.5	1024	+3.5M	+3.5M
Paramanu-Marathi	208M	1	-	26.40	30.97	-	-	88	1024	28.83B	28.83B
Paramanu-Marathi-instruct	208M	1	-	**26.93**	30.54	-	-	+0.5	1024	+2.5M	+2.5M
Paramanu-Hindi	367M	1	-	24.20	30.97	-	-	239	1024	66B	66B
Paramanu-Hindi-instruct	367M	1	-	25.54	**40.14**	-	-	+1	1024	+13M	+13M
Paramanu-Tamil	208M	1	-	-	-	33.34	-	112.5	1024	39.32B	39.32B
Paramanu-Tamil-instruct	208M	1	-	-	-	**34.80**	-	+0.5	1024	+3M	+3M
Paramanu-Telugu	208M	1	-	-	-	-	32.22	112.5	1024	39.32B	39.32B
Paramanu-Telugu-instruct	208M	1	-	-	-	-	**34.50**	+0.5	1024	+2.8M	+2.8M
IndicBART	244M	12	22.16	23.90	29.47	32.30	30.16	5,760	1024	18B	15.83B
BanglaT5	247M	1	21.92	-	-	-	-	3,000	512	196.6B	196.6B
BanglaByT5	300M	1	21.20	-	-	-	-	600	512	49.15B	49.15B
Bloom	560M	45	23.83	24.76	32.84	33.13	31.78	N/A	2048	350B	7.7B
Bloomz (instruction-tuned)	560M	45	24.01	25.29	32.31	33.30	32.25	N/A	2048	+13B	+1.4B
xGLM	564M	30	21.54	22.31	30.69	30.57	30.35	129,024	2048	500B	20B
Llama-3.2	1B	34	24.90	26.06	34.54	32.81	32.13	370,000	128,000	9T	N/A
TigerLLM (instruction-tuned)	1B	35	24.05	23.86	29.98	33.33	30.02	+216	32,768	+200M	+200M
Bloom	1.1B	45	24.75	25.78	33.87	32.94	33.04	N/A	2048	350B	7.7B
Bloomz (instruction-tuned)	1.1B	45	23.45	23.54	31.87	32.19	30.67	N/A	2048	+13B	+1.4B
mGPT	1.3B	61	22.86	23.30	31.95	29.73	30.96	86,016	2048	400B	15B
xGLM	1.7B	30	22.24	21.69	32.09	30.56	31.12	129,024	2048	500B	20B
Sarvam	2B	11	25.22	26.08	36.27	**35.25**	**34.55**	122,880	8192	4T	2T
Indic-Gemma-Navrasa	2B	16	25.27	26.02	34.40	33.73	33.55	+45	8192	$\geq 2T$	N/A
xGLM	2.9B	30	22.27	21.73	33.18	30.70	33.51	129,024	2048	500B	20B
Llama-3.2	3B	34	**30.17**	**31.36**	40.06	**36.59**	**34.86**	460,000	128,000	9T	N/A
Nemotron-Hindi	4B	15	-	**29.42**	**45.33**	-	-	N/A	4096	8.5T	491.2B
xGLM	4.5B	30	22.88	24.36	32.83	30.77	31.24	129,024	2048	500B	20B
Bloom	7B	45	25.47	25.61	35.92	33.95	33.28	N/A	2048	350B	7.7B
Bloomz (instruction-tuned)	7B	45	**37.77**	**37.68**	**40.80**	**41.41**	**39.71**	N/A	2048	+13B	+1.4B
OpenHathi (CPTLLama-2)	7B	28	-	25.40	35.41	-	-	N/A	4096	$\geq 2T$	$\geq 7B$
Airavata (instruction-tuned)	7B	28	-	26.64	36.93	-	-	N/A	4096	$\geq 2T$	N/A
xGLM	7.5B	30	22.99	22.47	34.16	30.73	31.73	129,024	2048	500B	20B
Llama-3	8B	34	**33.54**	**33.10**	**43.45**	**39.09**	**38.59**	1,300,000	8192	$\geq 15T$	N/A
Aya23	8B	23	25.76	28.69	**43.98**	34.12	31.02	N/A	8192	N/A	N/A

Table 2: Summary of Zero-Shot Benchmark Average Scores across Scripts (Bangla, Devanagari, Tamil, Telugu) and Languages. Models that performed better than our models are **underlined and bold**; the best performance of our model is in **bold**; '-' represents that monolingual and multilingual models are not evaluated on languages of different scripts which were not part of training but languages of same script were evaluated even if it was not part of training; '+' denotes additional tokens/training hours on top of pretrained models for instruction-tuning. Bloom reported a total training hours of 1,082,990 across several models.

models solely trained on texts written in Indian languages, as our models, which nevertheless remain competitive on these translated benchmark tasks.

4.4 Ablation Studies

Impact of tokenizer and data cleaning. Table 3 presents the impact of incorporating Unigram LM tokens into BPE and of data cleaning across five languages. Our tokenizer improves downstream task average performance over standard BPE by 1.91% (Bangla), 1.41% (Hindi), 1.53% (Marathi), 0.56% (Tamil), and 2.02% (Telugu). Additional data cleaning further boosts scores by 1% (Bangla), 3.5% (Marathi), 1% (Tamil), and 2% (Telugu), with a similar trend observed for BPE tokenizers underscoring the value of data preprocessing irrespective of tokenization strategy.

Impact of shrinking factor for position interpolation context window. Table 4 shows the impact of the position scaling factor α, at the basis of the position interpolation method used to accommodate more tokens than the permissible length. The average benchmark score monotonically increases with bigger context size for all languages except Telugu. Although we gain on larger sequences, on the flip side, interpolating token position indices to reside in a much narrower region might inject noise which may perturb language modeling benchmark performance.

5 Discussion

We discuss in this section several aspects of the Paramanu models, related to scaling, to the gains with instruction fine-tuning, to cross-lingual transfer and to N-shot degradation.

5.1 Scaling

In our study, smaller Hindi models (e.g., 162M) often outperformed larger ones (e.g., 367M) at fixed training durations, as shown in Appendix Ta-

Language	Configuration	MMLU	ARC	Belebele	XCOPA	XStoryCloze	HellaSwag	XNLI	Average
Bangla	full (ours)	**24.82**	**25.75**	**25.11**	-	-	-	-	**25.22**
	full w/o Unigram	22.66	23.61	23.67	-	-	-	-	23.31
	full w/o cleaning	23.67	24.21	25.00	-	-	-	-	24.29
	full w/o Unigram w/o cleaning	20.55	21.56	23.33	-	-	-	-	21.81
Hindi	full (ours)	**25.18**	**27.14**	**26.22**	-	**48.78**	**25.02**	**33.49**	**30.97**
	full w/o Unigram	23.35	25.54	25.22	-	46.78	24.89	**33.49**	29.87
	full w/o cleaning	24.72	22.25	25.44	-	48.16	24.06	32.34	29.49
	full w/o Unigram w/o cleaning	22.75	21.83	23.56	-	44.96	24.02	33.21	28.38
Marathi	full (ours)	**25.39**	**26.49**	**27.33**	-	-	-	-	**26.40**
	full w/o Unigram	25.31	23.20	26.11	-	-	-	-	24.87
	full w/o cleaning	22.47	22.42	24.00	-	-	-	-	22.96
	full w/o Unigram w/o cleaning	23.17	21.82	21.78	-	-	-	-	22.26
Tamil	full (ours)	**24.37**	**24.51**	**26.88**	**57.60**	-	-	-	**33.34**
	full w/o Unigram	23.87	24.08	26.11	57.00	-	-	-	32.76
	full w/o cleaning	23.25	23.38	26.56	56.20	-	-	-	32.35
	full w/o Unigram w/o cleaning	22.47	22.50	24.00	54.80	-	-	-	30.94
Telugu	full (ours)	**25.26**	**26.32**	**26.00**	-	**54.20**	-	-	**32.95**
	full w/o Unigram	25.12	21.75	24.44	-	52.42	-	-	30.93
	full w/o cleaning	24.16	22.68	23.11	-	53.73	-	-	30.92
	full w/o Unigram w/o cleaning	22.95	17.81	23.22	-	51.72	-	-	28.92

Table 3: Ablation study across Bangla (108M), Marathi (208M), Tamil (208M), Telugu (208M), and Hindi (367M) models. Evaluates the impact of tokenizer type and data cleaning. All scores are reported as zero-shot Accuracy (%). Dash (-) indicates benchmark not applicable or not available on LM-Eval.

bles 9, 10. With extended training, larger models (e.g., 367M) surpassed smaller counterparts (e.g., 162M), underscoring the need for size-appropriate training duration. Hence, we confirm prior findings (Hoffmann et al., 2022b; Kaplan et al., 2020) that larger models require proportionally more training to outperform smaller ones.

5.2 Instruction-tuning (IFT) Gains

Instruction tuning improves downstream performance across all five evaluated languages. Hindi shows the largest improvement (+9%), followed by Bangla (+4.3%). Gains are smaller for Tamil (+1.46%) and Telugu (+2.28%), likely due to lower-quality machine translation artifacts in instruction tuning datasets due to the lack of manual post-editing for these languages, which was not performed due to budget constraints. As one can note from Table 2, machine translation quality may also play a role here: poorer translation quality (Marathi vs Hindi) leads to lower IFT gains in our setting (models and datasets).

5.3 Cross-lingual Transfer

As shown in Table 2, the 208M Marathi model matches the 367M Hindi model on Hindi benchmarks (30.97), likely due to its lower perplexity (8.94 vs. 11.05), indicating effective cross-lingual transfer between Hindi and Marathi, which share the Devanagari script. Instruction tuning, however, is asymmetric: Marathi $\rightarrow$ Hindi slightly reduces performance (-0.43), while Hindi $\rightarrow$ Marathi improves it ($+1.34$) (Table 5). This suggests stronger transfer from high- (Hindi) to low-resource (Marathi) languages. Consequently, these monolingual models are suitable for cross-lingual downstream tasks such as translation and information retrieval. This observation aligns with prior works (Choenni et al., 2023; Faisal and Anastasopoulos, 2024; Zhang et al., 2025), showing that cross-lingual transfer in multilingual models is often directional and asymmetric, with stronger gains from high- to low-resource languages.

To further test how languages of Devanagari script generalize to unseen languages, we trained a Sanskrit monolingual model and a multilingual model, mParamanu-162M on languages of Indo-European family: Assamese, Bangla, Odia and three languages with Devanagari script, namely Sanskrit, Konkani, and Maithili. We intentionally kept out Hindi and Marathi from the training of our multilingual model to test its generalization to languages of Devanagari script. Table 5 shows strong zero-shot transfer within languages that share the same script, Devanagari. For example, Paramanu-Sanskrit 139M, trained without Hindi, achieves 31.05 avg on Hindi, outperforming xGLM 564M and approaching Bloom 560M. mParamanu-162M also generalizes well to Hindi and Marathi. This shows that both monolingual

Language	Configuration	MMLU	ARC	Belebele	XCOPA	XStoryCloze	HellaSwag	XNLI	Average
Bangla	α =1, ctx=256	22.62	24.04	23.00	-	-	-	-	23.22
	α =2, ctx=512	**25.84**	23.52	**25.44**	-	-	-	-	24.93
	α =3, ctx=768	23.39	**25.75**	25.33	-	-	-	-	24.82
	α =4, ctx=1024	24.82	**25.75**	25.22	-	-	-	-	**25.22**
Hindi	α =2, ctx=512	25.43	25.34	27.56	-	48.78	25.14	33.17	30.90
	α =3, ctx=768	25.36	25.86	26.44	-	47.78	25.13	33.29	30.64
	α =4, ctx=1024	25.18	27.14	26.22	-	48.78	25.02	33.49	30.97
Marathi	α =1, ctx=256	22.96	**28.48**	22.33	-	-	-	-	24.59
	α =2, ctx=512	24.45	26.06	25.89	-	-	-	-	25.46
	α =3, ctx=768	24.58	24.66	**27.78**	-	-	-	-	25.67
	α =4, ctx=1024	**25.39**	26.94	27.33	-	-	-	-	**26.55**
Tamil	α =1, ctx=256	22.72	**25.04**	23.22	53.20	-	-	-	31.04
	α =2, ctx=512	23.40	22.94	23.89	53.80	-	-	-	31.00
	α =3, ctx=768	**25.30**	23.91	23.33	54.80	-	-	-	31.84
	α =4, ctx=1024	24.37	24.51	**26.88**	**57.60**	-	-	-	**33.34**
Telugu	α =1, ctx=256	**26.83**	25.26	21.89	-	53.01	-	-	31.75
	α =2, ctx=512	25.06	**25.61**	**28.89**	-	53.34	-	-	**33.22**
	α =3, ctx=768	25.40	26.32	22.67	-	53.08	-	-	31.87
	α =4, ctx=1024	25.26	26.32	26.00	-	**54.20**	-	-	32.95

Table 4: Ablation study for shrinking factor α of position interpolation for varying context size (ctx) pretraining across Bangla (108M), Marathi (208M), Tamil (208M), Telugu (208M), and Hindi (367M) models for . All scores are reported as zero-shot Accuracy (%). Dash(-) indicates benchmark not applicable or not available on LM-Eval.

Model	Size	#langs	Devanagari	
			Marathi	Hindi
Paramanu-Sanskrit	139M	1	25.26	31.05
mParamanu	162M	6	25.28	30.07
Paramanu-Marathi	208M	1	26.40	30.97
Paramanu-Marathi-instruct	208M	1	26.93	30.54
Paramanu-Hindi	367M	1	24.20	30.97
Paramanu-Hindi-instruct	367M	1	25.54	40.14

Table 5: Average zero-shot benchmark scores for cross-lingual transfer among Devanagari languages (Hindi, Marathi, Sanskrit). Sanskrit and mParamanu were not trained on Hindi.

and multilingual models of shared script can be used for downstream cross-lingual tasks.

5.4 N-shot Degradation

The experiments and results reported in Table 13 in Appendix C.2 reveal a non-monotonic trend from 0-shot to 25-shot settings, with performance drops observed on XNLI-Hindi, XStoryCloze, and XCOPA. This behavior highlights that, for small pretrained models (< 400M parameters), adding more in-context examples does not necessarily yield improvements. Instead, effectiveness depends critically on the selection, sensitivity, quality, and formatting of the provided examples. While similar trends have been documented in larger models (Zhao et al., 2021; Liu et al., 2022; Mosbach et al., 2023), our findings provide empirical evidence that such inconsistencies also arise in small-scale Indic SLMs, where few-shot gains are often limited or unstable. One possible explanation is that in-context examples act as soft constraints that may inadvertently hinder generation when they are suboptimal or poorly aligned with the target task.

6 Conclusions

We introduce PARAMANU, the first family of open-source, Indian language-only sub-400M decoder language models trained from scratch on open-source, language-specific data. These models were designed to be broadly usable by NLP researchers. We show that, for low-resource, morphologically rich languages, small language-specific models (under 400M parameters and trained on fewer than 70B tokens) can outperform larger alternatives. *This suggests that, under constraints on model and data size, the optimal strategy is to build language-specific models with low-fertility, morphologically aligned tokenizers trained on cleaned data, rather than maximizing scale.* Our open-source models, tokenizers, and instruction-tuning datasets advance understanding of smaller LMs and provide a practical foundation for under-resourced researchers. We hope that they will enable further research on Indian languages.

In the future, we aim to scale our methodology to other morphologically rich and agglutinative languages.

Limitations

Building generative language models for Indian languages involves several challenges. Each stage such as data collection, pretraining, instruction tuning, and evaluation has its own limitations. Additionally, the societal and cultural implications of deploying AI in underrepresented linguistic communities are complex and beyond the full scope of this section.

Data. Our models are trained on a limited corpus consisting mainly of news articles, Wikipedia, and other structured sources in Indian languages, totaling only a few billion tokens. The dataset lacks diversity across key domains such as law, science, education, and general world knowledge. As a result, the models perform sub-optimally on benchmarks like MMLU, which test broad academic and professional understanding. As with any LLM, the data it is trained on determines the range and quality of its capabilities.

Training. Due to resource constraints, models were trained over multiple epochs on repeated tokens. Although this helps to reinforce learning in low-resource settings, it increases the risk of overfitting and can reduce generalization. Furthermore, our models were pretrained using a single A100 GPU, which significantly restricted training duration, batch size, and overall scale. As a result, the models may not have reached full convergence. The lack of compute and open-source infrastructure for Indian language models continues to be a major bottleneck. Detailed training logs are omitted due to space limitations.

Instruction Tuning and Safety. We instruction-tuned the models using 15,000 machine-translated instructions generated via Google Translate. This may introduce grammatical errors or semantic inconsistencies, as machine translation quality for Indic languages remains below human-level accuracy. No safety mechanisms such as prompt filtering, fact-checking, or toxicity detection were applied. The outputs shown in this paper are raw model responses, without post-processing. As a result, models may generate factually incorrect, biased, or inappropriate content. We emphasize the need for responsible use and future work on safety and alignment.

Evaluation. Evaluation was conducted in a zero-shot setting without task-specific fine-tuning. This limits performance, especially on complex or domain-specific tasks. Benchmarks like MMLU highlight the impact of limited training data and the absence of instruction tuning aligned with task objectives. Future improvements can be expected through more diverse data and supervised fine-tuning.

Ethical Considerations

In this work, we advocate for greater openness in developing generative language models, especially for low-resource morphologically rich Indian languages serving more than 1 billion speakers. Open access is crucial for deepening our scientific understanding of these models and ensuring that communities beyond the Global North can actively participate in their advancement. Training on openly available datasets not only supports transparency but also helps bridge the gap for languages and regions that have historically been underrepresented in AI research.

By releasing our models and tokenizers openly, we empower researchers, developers, and communities to build upon existing work rather than starting from scratch saving resources and reducing environmental impact. While we acknowledge the risks of misuse, we believe that broader access enables more diverse efforts to identify, study, and mitigate potential harms. We recognize that openness comes with risks these models could be misused. However, we believe open access also helps researchers identify and reduce such risks more effectively by encouraging diverse solutions.

Acknowledgments

This work was partly supported by Gyan AI Research, the ANR GUIDANCE project, grant ANR-23-IAS1-0003 of the French Agence Nationale de la Recherche, and the Institut Universitaire de France (IUF). This work was partly performed using HPC resources from GENCI-IDRIS (Grant 2025-AD011016103).

References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models.

Duarte Miguel Alves, José Pombal, Nuno M Guerreiro, Pedro Henrique Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and Andre Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. In *First Conference on Language Modeling*.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, Kelly Marchisio, Max Bartolo, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Aidan Gomez, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. Aya 23: Open weight releases to further multilingual progress.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.

Scott Barnett, Zac Brannelly, Stefanus Kurniawan, and Sheng Wong. 2024. Fine-tuning or fine-failing? debunking performance myths in large language models.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, and Rifat Shahriyar. 2023. BanglaNLG and BanglaT5: Benchmarks and resources for evaluating low-resource natural language generation in Bangla. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 726–735, Dubrovnik, Croatia. Association for Computational Linguistics.

Pramit Bhattacharyya and Arnab Bhattacharya. 2025. BanglaByT5: Byte-level modelling for Bangla. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 5551–5560, Suzhou, China. Association for Computational Linguistics.

Pramit Bhattacharyya, Joydeep Mondal, Subhadip Maji, and Arnab Bhattacharya. 2023. VACASPATI: A diverse corpus of Bangla literature. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1118–1130, Nusa Dua, Bali. Association for Computational Linguistics.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. Gpt-neox-20b: An open-source autoregressive language model.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online. Association for Computational Linguistics.

Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Ben Bergen. 2024. When is multilinguality a curse? language modeling for 250 high- and low-resource languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4074–4096, Miami, Florida, USA. Association for Computational Linguistics.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023. Extending context window of large language models via positional interpolation.

Rochelle Choenni, Dan Garrette, and Ekaterina Shutova. 2023. How do languages influence each other? studying cross-lingual data sharing during LM fine-tuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13244–13257, Singapore. Association for Computational Linguistics.

Monojit Choudhury. 2023. Generative ai has a language problem. *Nature human behaviour*, 7(11):1802–1803.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sashank Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2024. Palm: scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24(1).

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Chaitali Dangarikar, Arnab Bhattacharya, Karthika N J, Chaitanya S Lakkundi, Ganesh Ramakrishnan, Annarao Kulkarni, Shivani V, Pramit Bhattacharyya, and Hrishikesh Terdalkar. 2024. Samanvaya: An interlingua for the unity of indian languages. *Central Sanskrit University*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Toronto, Canada. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katay-

oun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models.

Fahim Faisal and Antonios Anastasopoulos. 2024. An efficient approach for studying cross-lingual transfer in multilingual language models. In *Proceedings of the Fourth Workshop on Multilingual Representation Learning (MRL 2024)*, pages 45–92, Miami, Florida, USA. Association for Computational Linguistics.

Jay Gala, Thanmay Jayakumar, Jaavid Aktar Hussain, Aswanth Kumar M, Mohammed Safi Ur Rahman Khan, Diptesh Kanojia, Ratish Puduppully, Mitesh M. Khapra, Raj Dabre, Rudra Murthy, and Anoop Kunchukuttan. 2024. Airavata: Introducing hindi instruction-tuned llm.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, MarcAurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin

Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama

3 herd of models.

Yanzhu Guo, Simone Conia, Zelin Zhou, Min Li, Saloni Potdar, and Henry Xiao. 2024a. Do large language models have an english accent? evaluating and improving the naturalness of multilingual llms.

Yanzhu Guo, Guokan Shang, and Chloé Clavel. 2024b. Benchmarking linguistic diversity of large language models. *ArXiv*, abs/2412.10271.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, Oriol Vinyals, Jack Rae, and Laurent Sifre. 2022a. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pages 30016–30030. Curran Associates, Inc.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022b. Training compute-optimal large language models.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Shaoxiong Ji, Zihao Li, Indraneil Paul, Jaakko Paavola, Peiqin Lin, Pinzhen Chen, Dayyán O'Brien, Hengyu Luo, Hinrich Schuetze, Jörg Tiedemann, and Barry Haddow. 2025. EMMA-500: Enhancing massively multilingual adaptation of large language models.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Raviraj Joshi, Kanishk Singla, Anusha Kamath, Raunak Kalani, Rakesh Paul, Utkarsh Vaidya, Sanjay Singh Chauhan, Niranjan Wartikar, and Eileen Long. 2025. Adapting multilingual LLMs to low-resource languages using continued pre-training and synthetic corpus: A case study for Hindi LLMs. In *Proceedings of the First Workshop on Natural Language Processing for Indo-Aryan and Dravidian Languages*, pages 50–57, Abu Dhabi. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. IndicLLMSuite: A blueprint for creating pre-training and fine-tuning datasets for Indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879, Bangkok, Thailand. Association for Computational Linguistics.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates.

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.

Viet Dac Lai, Nghia Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023a. ChatGPT beyond English: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13171–13189, Singapore. Association for Computational Linguistics.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023b. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Haokun Liu, Derek Tam, Muqeeth Mohammed, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*.

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

Yinquan Lu, Wenhao Zhu, Lei Li, Yu Qiao, and Fei Yuan. 2024. LLaMAX: Scaling linguistic horizons of LLM by enhancing translation capabilities beyond 100 languages. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10748–10772, Miami, Florida, USA. Association for Computational Linguistics.

Yash Madhani, Mitesh M. Khapra, and Anoop Kunchukuttan. 2023. Bhasa-Abhijnaanam: Native-script and romanized language identification for 22 Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 816–826, Toronto, Canada. Association for Computational Linguistics.

Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and mitigating language confusion in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6653–6677, Miami, Florida, USA. Association for Computational Linguistics.

Meta AI. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/. Accessed: 2025-12-27.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12284–12314, Toronto, Canada. Association for Computational Linguistics.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkadir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Murhabazi Espoir, Elan van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Chinenye Emezue, Bonaventure F. P. Dossou, Blessing Sibanda, Blessing Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. 2020. Participatory research for low-resourced machine translation: A case study in African languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160, Online. Association for Computational Linguistics.

Hellina Hailu Nigatu, Atnafu Lambebo Tonja, Benjamin Rosman, Thamar Solorio, and Monojit Choudhury. 2024. The zeno's paradox of 'low-resource' languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17753–17774, Miami, Florida, USA. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2021. UNKs everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10186–10203, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jan Pfister, Julia Wunderle, and Andreas Hotho. 2025. LLäMmlein: Transparent, compact and competitive

German-only language models from scratch. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2227–2246, Vienna, Austria. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Anselm Levskaya, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. 2022. Efficiently scaling transformer inference. *ArXiv*, abs/2211.05102.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Nishat Raihan and Marcos Zampieri. 2025. TigerLLM - a family of Bangla large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 887–896, Vienna, Austria. Association for Computational Linguistics.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Sarvam2B. 2024. sarvamai/sarvam-2b-v0.5 ů Hugging Face — huggingface.co. https://huggingface.co/sarvamai/sarvam-2b-v0.5. [Accessed 15-09-2024].

SarvamAI. 2023. Openhathi series. Accessed: 2025-01-12.

Lisa Schut, Yarin Gal, and Sebastian Farquhar. 2025. Do multilingual LLMs think in english? In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*.

Lane Schwartz. 2022. Primum Non Nocere: Before working with Indigenous data, the ACL must confront ongoing colonialism. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 724–731, Dublin, Ireland. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sheikh Shafayat, Eunsu Kim, Juhyun Oh, and Alice Oh. 2024. Multi-fact: Assessing factuality of multilingual llms using factscore.

Noam Shazeer. 2020. Glu variants improve transformer.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Anastasia Kozlova, Vladislav Mikhailov, and Tatiana Shavrina. 2024. mGPT: Few-shot learners go multilingual. *Transactions of the Association for Computational Linguistics*, 12:58–79.

Anil Kumar Singh. 2008. Natural language processing for less privileged languages: Where do we come from? where are we going? In *International Joint Conference on Natural Language Processing*.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2022. Roformer: Enhanced transformer with rotary position embedding.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Lintang Sutawika, Hailey Schoelkopf, Leo Gao, Stella Biderman, Baber Abbasi, Jonathan Tow, ben fattori, Charles Lovering, farzanehnakhaee70, Jason Phang, Anish Thite, Fazz, Aflah, Niklas Muennighoff, Thomas Wang, sdtblck, gakada, nopperl, researcher2, tttyuntian, Chris, Julen Etxaniz, Zdenk Kasner, Khalid, Jeffrey Hsu, Hanwool Albert Lee, Anjor Kanekar, AndyZwei, Pawan Sasanka Ammanamanchi, and Dirk Groeneveld. 2024. Eleutherai/lm-evaluation-harness: v0.4.1.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogoziska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Pluciska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Telugu-LLM-Labs. 2025. Indic-gemma-2b-finetuned-sft-navarasa-2.0. Accessed: 2025-03-09.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Yulia Tsvetkov. 2017. Opportunities and challenges in working with low-resource languages. *Slides Part-1*, 2.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 60006010, Red Hook, NY, USA. Curran Associates Inc.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ili, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González

Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario ako, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdenk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model.

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.

Greg Yang, Edward J Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. 2021. Tuning large neural networks via zero-shot hyperparameter transfer. In *Advances in Neural Information Processing Systems*.

Biao Zhang and Rico Sennrich. 2019. *Root mean square layer normalization*. Curran Associates Inc., Red Hook, NY, USA.

Chen Zhang, Zhiyuan Liao, and Yansong Feng. 2025. Cross-lingual transfer of cultural knowledge: An asymmetric phenomenon. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 147–157, Vienna, Austria. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

Tony Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*.

Wenzhen Zheng, Wenbo Pan, Xu Xu, Libo Qin, Li Yue, and Ming Zhou. 2024a. Breaking language barriers: Cross-lingual continual pre-training at scale. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7725–7738, Miami, Florida, USA. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024b. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

Appendix

A Tokenizers

We provide here Table 6, fully analyzed and discussed in Section 3.3.

B Annotators Information and Guidelines to design Bangla Instruction-Tuning Dataset

The human-authored Bangla 5K instruction dataset was created by 20 native Bangla-speaking annotators, each responsible for generating 250 instruction-response pairs, resulting in 5,000 high-quality samples. All annotators have prior experience in linguistic annotation, translation, or writing. Each instruction-response pair underwent peer review for linguistics and factual correctness, fluency, clarity, and cultural appropriateness. Agreement was measured using Fleiss Kappa across all annotators, yielding an average score of 0.87, indicating strong reliability and consistency in annotation. Due to budget constraints, manual post-editing was performed only for Hindi and Marathi. Word error rates averaged 8% for Hindi and 13% for Marathi, and these errors were manually corrected by annotators following grammatical correction guidelines.

We construct a manually curated dataset of 5,000 high-quality instructionresponse pairs in Bangla, focusing on cultural, literary, and practical domains relevant to Bangla-speaking users in India. Below, we outline the core annotation guidelines used to ensure consistency, linguistic accuracy, and cultural relevance.

Annotation Protocol

Each data point consists of:

- **Instruction:** A natural prompt in Bangla, emulating user queries or tasks.
- **Response:** A helpful, accurate, and contextually appropriate answer in Bangla.
- **Category (optional):** Domain label (e.g., Literature, Daily Life).

Domain Coverage.

To promote diversity, we target a balanced distribution across ten culturally salient domains: literature, culture/tradition, history/politics, religion, education, health and daily life, technology, ethics, creative writing, and casual conversation.

Instruction Types.

We include a variety of prompt formats, including:

Text	BPE	BPE+Unigram
किंवदंति (kiṃbadanti)	['_किं', 'व', 'द', 'ं', 'ति']	['_किं', 'व', 'द', 'ंति']
नियमानुवर्तिता (niyamānubartitā)	:['_निय', 'मान', 'ु', 'वर्त', 'िता']	['_निय', 'मा', 'नु', 'वर्त', 'िता']
परिवारेर (paribārera)	['_परि', 'वारेर']	['_परिवार', 'ेर']
স্বাচ্ছন্দ্য (sbācchandya)	:['_স্বা', 'চ্ছ', 'ন্দ', '্য']	:['_স্বা', 'চ্ছ', 'ন্দ', '্য']
व्यक्तिगत (byaktigata)	:['_व्यक्ति', 'गत']	:['_व्यक्ति', 'गत']
பயணித்தார்கள் (payaṇittārkaḷ)	['_பயணி', 'த்த', 'ார்கள்']	['_பயணித்த', 'ார்கள்']
எழுதிக்கொண்டேபோயிருந்தார்கள் (ĕḻutikkŏṇṭepoyiruntārkaḷ)	['_எழு', 'திக்க', 'ொண்ட', 'ே', 'போ', 'ய', 'ிருந்த', 'ார்கள்']	['_எழு', 'திக்க', 'ொண்ட', 'ே', 'போ', 'யிருந்த', 'ார்கள்']
చదువుకుంటూనేవున్నవారికోసం (caduvukuṃṭūnevunnavārikosaṃ)	['_చదువు', 'కు', 'ంటూ', 'నే', 'వు', 'న్న', 'వార', 'ికో', 'సం']	['_చదువు', 'కు', 'ంటూ', 'నే', 'వ', 'ున్న', 'వార', 'ికో', 'సం']
लिहूनघेतलेल्यानुसारही (lihūnaghetalelyānusārahī)	['_लि', 'हून', 'घ', 'ेत', 'लेल्या', 'नुसार', 'ही']	['_लिह', 'ून', 'घे', 'तले', 'ल्या', 'नुसार', 'ही']
लिखकरदेनेवालोंकेलिएभी (likhakaradenevālōṃkeliebhī)	['_लिखकर', 'दे', 'ने', 'वालों', 'के', 'लिए', 'भी']	['_लिख', 'कर', 'दे', 'ने', 'वाले', 'ओं', 'के', 'लिए', 'भी']

Table 6: Comparison of BPE and BPE+Unigram Tokenization for 5 Indian languages.

Language	Avg. Instruction Length	Avg. Input. Length	Avg. Response Length
Bangla	63.7 ± 61.47	95.36 ± 468.49	505.45 ± 499.93
Hindi	63.54 ± 63.28	97.31 ± 478.18	461.89 ± 462.19
Marathi	58.91 ± 65.08	95.73 ± 470.83	499.30 ± 478.70
Tamil	72.72 ± 66.83	109.33 ± 534.16	496.96 ± 456.53
Telugu	64.69 ± 63.07	99.82 ± 488.04	458.20 ± 419.08

Table 7: Average # tokens in each Instruction, Input, and Response across 5 Indian languages for the instruction-tuning datasets.

- Informational Q&A (e.g., historical facts, literary interpretation)
- How-to and procedural instructions
- Creative writing prompts (e.g., poem, story)
- Comparative and opinion-based prompts
- Grammar correction and translation tasks

Language Guidelines.

Instructions and responses are written in standard colloquial Bangla. Annotators avoid code-switching and maintain gender neutrality unless contextually necessary. All content is free of offensive, biased, or politically sensitive material.

Quality Control

All samples undergo dual-stage validation:

1. Peer review for factual correctness, fluency, and clarity.
2. Final check for linguistic naturalness and domain balance.

This dataset aims to support instruction-tuning of LLMs in Bangla by providing diverse, culturally grounded, and linguistically accurate examples.

C Added Evaluation

C.1 Language Specific Evaluation

In this subsection, we discuss language-specific benchmarks for five Indian languages: Bangla, Hindi, Marathi, Tamil, and Telugu. Tables 8, 9, 10, 11, 12 show the respective languauge specific quantitative benchmark results.

Bangla. Table 8 shows that Paramanu-Bangla 108M outperforms 13 of 15 LLMs in the 500M–3B range, tying with Sarvam 2B despite being 18× smaller. Although it trails 5 of 7 LLMs above 3B, it surpasses xGLM 7.5B and Bloom 1.1B in average score across the MMLU, ARC, and Belebele benchmarks, while being 70× smaller than xGLM 7.5B and pretrained on only 26.21 billion tokens. Trained exclusively on Bangla literature and Wikipedia, its limited domain coverage likely accounts for its weaker MMLU performance. Instruction tuning on 27k Bangla instructions produces PARAMANU-BANGLA-INSTRUCT 108M, which surpasses all 15 LLMs in the 500M–3B range and 4 of 7 models above 3B, including xGLM 4.5B, Bloom 7B, and xGLM 7.5B. On ARC, it outperforms all models. The multilingual variant (MPARAMANU) underperforms its monolingual counterpart, highlighting the trade-off associated with multilinguality. Notably, Bloom 1.1B-Instruct also exhibits a 1.3% drop in Bangla performance.

Devanagari. Table 10 and Table 9 show that PARAMANU models achieve strong results on Devanagari benchmarks despite their small size and limited pretraining. PARAMANU-MARATHI 208M outperforms all LLMs in the 500M-3B range and 4 of 10 models above 3B on Marathi. After instruction-tuning, PARAMANU-MARATHI 208M outperforms 7 of 10 models above 3B. MPARAMANU 162M and monolingual PARAMANU-SANSKRIT, despite not being trained on Hindi and Marathi data, surpass the random baseline and 10 of 23 LLMs via cross-lingual transfer. PARAMANU-HINDI-INSTRUCT 367M, tuned on 27k Hindi instructions and a 52k synthetic Alpaca machine-translated Hindi dataset, exceeds 11 of 23 LLMs on Marathi and, overall, 19 of 23 LLMs on Hindi. It outperforms all models (500M–3B) and 6 of 10 models above 3B on Hindi. Though Nemotron-Hindi 4B and Aya23 8B lead in Hindi, our models are significantly more efficient. Unlike costly continual pretraining and fine-tuning of LLMs (e.g., OpenHathi 7B, Airavata 7B, Indic-Gemma 2B), our pretrain from scratch approach of tiny monolingual models for low-resource languages with language-specific tokenization yields

Models	Size	MMLU-Bangla	ARC-Bangla	Belebele-Bangla	Average (Bangla)	Belebele-Assamese
Paramanu-Bangla (ours)	108M	24.82	25.75	25.11	25.22	25.33
Paramanu-Bangla-instruct (ours)	108M	**27.60**	**28.50**	**32.45**	**29.52**	**30.54**
mParamanu (ours)	92M*	25.78	26.18	22.44	24.80	22.44
mParamanu (ours)	162M	25.29	26.01	27.44	26.24	29.00
mParamanu (ours)	237M*	22.52	25.92	24.44	24.29	24.28
IndicBART	244M	26.49	17.79	22.22	22.16	22.11
BanglaT5	247M	22.62	20.27	22.89	21.92	22.89
BanglaByT5	300M	24.23	17.37	22.00	21.20	21.78
Bloom	560M	22.61	26.00	22.89	23.83	22.78
Bloomz (instruction-tuned)	560M	25.82	23.43	22.77	24.01	25.11
xGLM	564M	22.61	18.47	23.55	21.54	22.77
Llama-3.2	1B	25.83	20.44	28.44	24.90	28.33
TigerLLM (instruction-tuned)	1B	24.99	19.16	28.00	24.05	28.11
Bloom	1.1B	23.90	24.37	26.00	24.75	26.89
Bloomz	1.1B	25.19	20.61	24.55	23.45	22.33
mGPT	1.3B	24.12	18.90	25.55	22.86	24.88
xGLM	1.7B	24.46	19.50	22.77	22.24	22.55
Sarvam	2B	24.05	28.40	23.22	25.22	27.78
Indic-Gemma-Navrasa (instruction-tuned)	2B	**29.07**	20.87	25.88	25.27	24.44
xGLM	2.9B	23.75	19.85	23.22	22.27	22.66
Llama-3.2	3B	**32.51**	21.47	**36.55**	**30.17**	**33.88**
xGLM	4.5B	25.74	21.04	21.88	22.88	22.44
Bloom	7B	27.10	26.09	23.22	25.47	23.11
Bloomz (instruction-tuned)	7B	**32.46**	27.20	**53.67**	**37.77**	**48.00**
xGLM	7.5B	24.69	19.41	24.88	22.99	24.44
Llama-3	8B	**35.77**	23.86	**41.00**	**33.54**	**35.77**
Aya23	8B	26.94	18.81	31.55	25.76	**31.22**

* Trained for the same number of steps.

Table 8: Zero-shot evaluation of LLMs across translated benchmarks of MMLU, HellaSwag, ARC datasets, and Belebele in Bangla script. All benchmarks report Accuracy. Models that performed better than our models have been **underlined and bold**, the best performance of our model has been **bold**.

Models	Size	MMLU-Hindi	HellaSwag-Hindi	ARC-Hindi	XStoryCloze-Hindi	XNLI-Hindi	Belebele-Hindi	Average
Paramanu-Sanskrit (ours)	139M	25.16	25.64	25.17	50.23	34.46	25.66	31.05
mParamanu (ours)	92M*	23.94	25.31	26.28	47.05	33.45	24.44	30.07
mParamanu (ours)	162M*	24.84	24.87	22.35	49.24	33.70	25.44	30.07
mParamanu (ours)	237M*	22.78	25.17	27.57	46.79	33.13	21.89	29.55
Paramanu-Marathi (ours)	208M	25.49	26.59	23.97	48.71	33.73	27.33	30.97
Paramanu-Marathi-instruct (ours)	208M	23.71	27.78	23.89	50.89	34.10	22.89	30.54
Paramanu-Hindi (ours)	162M*	23.15	25.37	27.31	48.91	33.17	22.67	30.09
IndicBART	244M	25.37	25.86	19.94	49.56	33.33	22.77	29.47
Paramanu-Hindi (ours)	367M*	24.38	24.83	27.05	47.92	32.00	23.33	29.92
Paramanu-Hindi (ours)	367M	25.18	25.02	27.14	48.78	33.49	26.22	30.97
Paramanu-Hindi-instruct (ours)	367M	**30.25**	**29.42**	**30.23**	**58.00**	**40.25**	**42.78**	**40.14**
Bloom	560M	23.67	27.50	23.88	54.79	40.80	26.44	32.84
Bloomz (instruction-tuned)	560M	25.87	26.48	24.40	55.53	35.58	26.00	32.31
xGLM	564M	22.70	26.96	20.20	52.00	38.31	24.00	30.69
Llama-3.2	1B	28.22	28.95	23.97	56.25	40.08	29.77	34.54
TigerLLM (instruction-tuned)	1B	24.25	25.43	20.20	50.56	33.05	26.44	29.98
Bloom	1.1B	23.86	28.28	24.74	55.59	**42.77**	28.00	33.87
Bloomz	1.1B	24.90	28.54	21.06	56.65	37.87	22.22	31.87
mGPT	1.3B	24.26	27.42	19.86	52.74	**41.32**	26.11	31.95
xGLM	1.7B	24.70	28.46	20.63	55.79	38.99	24.00	32.09
Sarvam	2B	24.54	**33.66**	28.00	**60.29**	**46.74**	24.44	36.27
Indic-Gemma-Navrasa (instruction-tuned)	2B	29.63	**30.31**	22.43	**60.62**	37.22	26.22	34.40
xGLM	2.9B	24.47	29.19	21.23	57.57	**42.65**	24.00	33.18
Llama-3.2	3B	**34.88**	**32.78**	24.65	**60.75**	**43.45**	**43.88**	40.06
Nemotron-Hindi	4B	**41.64**	**37.86**	**31.93**	**65.91**	**40.92**	**53.77**	**45.33**
xGLM	4.5B	25.93	28.44	21.06	56.84	**41.28**	23.44	32.83
Bloom	7B	27.04	**31.39**	26.36	**60.55**	**47.18**	23.00	35.92
Bloomz (instruction-tuned)	7B	**35.55**	28.57	29.36	57.71	**40.52**	**53.11**	**40.80**
OpenHathi (Llama-2 CPT)	7B	27.69	**30.54**	25.51	57.04	39.03	32.66	35.41
Airavata (instruction-tuned OpenHathi)	7B	**30.43**	**29.53**	25.60	55.59	39.04	41.44	36.93
xGLM	7.5B	26.27	**30.52**	21.40	**58.70**	**45.74**	22.33	34.16
Llama-3	8B	**40.05**	**35.48**	27.48	**63.07**	**45.30**	**49.33**	**43.45**
Aya23	8B	**33.68**	**36.24**	29.88	**64.39**	**47.18**	**52.55**	**43.98**

* Trained for the same number of steps.

Table 9: Zero-shot evaluation of LLMs for cross-lingual language transfer in Hindi. All benchmarks report Accuracy. Models that performed better than our models have been underlined and bold, the best performance of our model has been **bold**.

Models	Size	MMLU-Marathi	ARC-Marathi	Belebele-Marathi	Average
Paramanu-Sanskrit (ours)	139M	24.96	26.49	24.33	25.26
mParamanu (ours)	92M*	23.37	27.53	25.22	25.37
mParamanu (ours)	162M	25.68	22.16	**28.00**	25.28
mParamanu (ours)	237M*	22.73	24.16	23.67	23.52
Paramanu-Marathi (ours)	208M	25.39	26.49	27.33	26.40
Paramanu-Marathi-instruct (ours)	208M	**25.97**	**26.94**	27.87	**26.93**
Paramanu-Hindi (ours)	162M	22.83	24.76	22.78	23.45
IndicBART	244M	25.73	23.11	22.88	23.90
Paramanu-Hindi (ours)	367M	23.78	24.16	24.66	24.20
Paramanu-Hindi-instruct (ours)	367M	24.54	25.63	26.44	25.54
Bloom	560M	22.78	24.50	27.00	24.76
Bloomz (instruction-tuned)	560M	26.20	24.24	25.44	25.29
xGLM	564M	22.53	21.29	23.11	22.31
Llama-3.2	1B	26.37	23.72	**28.11**	26.06
TigerLLM (instruction-tuned)	1B	23.98	22.16	25.44	23.86
Bloom	1.1B	23.93	25.10	**28.33**	25.78
Bloomz (instruction-tuned)	1.1B	24.82	21.81	24.0	23.54
mGPT	1.3B	23.26	22.33	24.33	23.30
xGLM	1.7B	22.54	20.43	22.11	21.69
Sarvam	2B	23.96	27.53	26.77	26.08
Indic-Gemma-Navrasa (instruction-tuned)	2B	28.29	22.77	27.00	26.02
xGLM	2.9B	23.10	19.56	22.55	21.73
Llama-3.2	3B	**31.83**	24.93	**37.33**	**31.36**
Nemotron-Hindi	4B	**32.32**	23.72	**32.22**	29.42
xGLM	4.5B	25.59	23.29	24.22	24.36
Bloom	7B	27.30	25.54	24.00	25.61
Bloomz (instruction-tuned)	7B	**32.62**	**27.44**	**53.00**	**37.68**
OpenHathi (Llama-2 CPT))	7B	26.09	24.24	25.88	25.40
Airavata (instruction-tuned of OpenHathi)	7B	26.15	23.90	**29.89**	26.64
xGLM	7.5B	23.73	21.91	21.77	22.47
Llama-3	8B	**35.47**	25.19	**38.66**	**33.10**
Aya23	8B	27.80	22.94	**35.33**	28.69

* Trained for the same number of steps.

Table 10: Zero-shot evaluation of LLMs for cross-lingual language transfer in Marathi. All benchmarks report Accuracy. Models that performed better than our models have been underlined and bold, the best performance of our model has been **bold**.

Models	Size	Belebele-Tamil	XCOPA-Tamil	MMLU-Tamil	ARC-Tamil	Average
Paramanu-Tamil (ours)	208M	26.88	**57.60**	24.37	24.51	33.34
Paramanu-Tamil-instruct (ours)	208M	**30.22**	56.00	**26.95**	**26.04**	**34.80**
IndicBART	244M	27.11	55.00	25.39	21.71	32.30
Bloom	560M	27.22	55.80	23.95	25.57	33.13
Bloomz (instruction-tuned)	560M	23.55	**58.60**	25.78	25.30	33.30
xGLM	564M	22.77	56.20	22.74	20.57	30.57
Llama-3.2	1B	28.44	55.60	25.95	21.27	32.81
TigerLLM (instruction-tuned)	1B	27.44	**59.80**	24.42	21.54	33.33
Bloom	1.1B	25.77	57.00	24.67	24.34	32.94
Bloomz	1.1B	22.66	57.40	26.10	22.59	32.19
mGPT	1.3B	20.88	53.20	23.50	21.36	29.73
xGLM	1.7B	21.88	55.00	23.66	21.71	30.56
Sarvam	2B	27.44	**63.00**	24.06	**26.53**	**35.25**
Indic-Gemma-Navrasa (instruction-tuned)	2B	25.44	**59.00**	**27.84**	22.67	33.73
xGLM	2.9B	23.44	54.20	24.33	20.84	30.70
Llama-3.2	3B	**34.00**	**59.00**	**29.47**	23.90	**36.59**
xGLM	4.5B	22.66	55.20	24.03	21.19	30.77
Bloom	7B	25.55	**59.20**	26.39	24.69	33.95
Bloomz (instruction-tuned)	7B	**50.66**	57.40	**29.48**	**28.10**	**41.41**
xGLM	7.5B	22.44	54.40	24.39	21.71	30.73
Llama-3	8B	**38.55**	**59.80**	**31.66**	**26.35**	**39.09**
Aya23	8B	**33.55**	55.60	26.14	21.19	34.12

Table 11: Zero-shot evaluation of LLMs in Tamil script models. All benchmarks report Accuracy. Models that performed better than our models have been underlined and bold, the best performance of our model has been **bold**.

Models	Size	Belebele-Telugu	XStoryCloze-Telugu	MMLU-Telugu	ARC-Telugu	Average
Paramanu-Telugu (ours)	208M	26.00	51.42	25.12	**26.32**	32.22
Paramanu-Telugu-instruct (ours)	208M	**27.50**	**58.00**	**26.75**	25.75	**34.50**
IndicBART	244M	26.88	48.31	25.11	20.35	30.16
Bloom	560M	23.55	55.65	24.10	23.85	31.78
Bloomz (instruction-tuned)	560M	22.44	54.86	**26.82**	24.91	32.25
xGLM	564M	25.11	55.85	22.91	17.54	30.35
Llama-3.2	1B	**28.44**	54.86	25.40	19.82	32.13
TigerLLM (instruction-tuned)	1B	24.66	48.77	24.29	22.36	30.02
Bloom	1.1B	26.88	56.38	24.53	24.38	33.04
Bloomz (instruction-tuned)	1.1B	23.11	55.32	25.49	18.77	30.67
mGPT	1.3B	22.88	57.25	25.85	17.89	30.96
xGLM	1.7B	23.66	**58.23**	24.34	18.24	31.12
Sarvam	2B	**27.66**	**60.09**	24.67	25.78	**34.55**
Indic-Gemma-Navrasa (instruction-tuned)	2B	26.55	**58.57**	**28.58**	20.52	33.55
xGLM	2.9B	22.66	**60.09**	23.45	18.85	33.51
Llama-3.2	3B	**31.55**	**58.17**	**29.47**	20.26	**34.86**
xGLM	4.5B	23.66	57.04	24.87	19.38	31.24
Bloom	7B	24.66	57.37	26.62	24.47	33.28
Bloomz (instruction-tuned)	7B	**43.11**	**58.23**	**29.55**	**27.98**	**39.71**
xGLM	7.5B	24.66	**60.22**	23.90	18.15	31.73
Llama-3	8B	**36.88**	**63.53**	**32.74**	21.22	**38.59**
Aya23	8B	**28.55**	54.07	19.91	21.57	31.02

Table 12: Zero-shot evaluation of LLMs in Telugu script models. All benchmarks report Accuracy. Models that performed better than our models have been underlined and bold, the best performance of our model has been **bold**.

competitive results using just 240 GPU-hours.

Tamil. From Table 11, our model, Paramanu-Tamil (208M), outperformed 11 of 13 in range 500M-3B including Bloom, Llama-3.2, mGPT, xGLM) across four benchmarks in Tamil, coming close to Sarvam (2B) despite being 10 times smaller and trained on 76 times less Indian tokens compared to Sarvam 2B. However, its performance on MMLU is lower than the random baseline like many LLMs including Sarvam 2B in comparison as shown in the table as the model is mostly trained on Tamil news corpora. Paramanu-Tamil-instruct which is instruction-tuned on our translated 23,000 instructions dataset performed better than all models except Sarvam 2B in the range between 500M-3B and 4 LLMs out of 7 in the range of 3B and 8B.

Telugu. From Table 12, Paramanu-Telugu 208M outperformed 8 models out of 13 up to 3B and 3 models (XGLM 4.5B, 7.5B, Aya23 8B) out of 7 between 3B and 8B despite being trained Telugu pretrained on 39.32 billion tokens. After instruction-tuning on 23,000 machine translated instructions, Paramanu-Telugu-instruct (208M) outperformed all models between 500M and 3B except Sarvam 2B in the range of 500M and 3B and underperformed than 3 models of 7 in the range of 3B and 8B. On ARC benchmark, our models peformed better than all models except Bloomz 7B. The improvements in metric scores for Tamil and Telugu instruction-tuned models were modest, likely due to lower-quality machine translations from Bangla compared to Hindi.

C.2 n-shot degradation

As discussed and analyzed in Section 5.4, in this subsection we present Table 13, which shows n-shot performance (0-shot, 5-shot, and 25-shot) on language benchmarks.

C.3 Human Evaluation

Automated evaluation metrics may overlook significant qualitative enhancements, especially when model outputs align well with particular linguistic or cultural contexts (Barnett et al., 2024). We thus performed human evaluation for Bangla and Hindi for all pretrained models, asking 10 annotators to evaluate top-3 responses for 10 prompts on a scale of 0 (worst) to 5 (best). We reached inter-annotator kappa score of 0.85 for Bangla, and 0.79 for Hindi. Figure 3 shows average human ratings for text generation across four standard dimensions: grammar, coherence, creativity, and factuality. As one can note, Paramanu outperforms all other models on Bangla, and all other models but Llma3-8B and Aya23-8B on Hindi, which have comparable performance to Paramanu.

C.4 Perplexity, MFU, CPU Inference Speed

Table 14 lists the test perplexity, MFU metric during pretraining and CPU inference speed of our various pretrained models. In terms of quantitative evaluation of language modeling, the lower the perplexity, the better is the language model. As one can note, perplexity generally improves with increasing model scale, consistent with neural scaling laws (Kaplan et al., 2020; Hoffmann et al., 2022b), though the trend is not strictly monotonic across languages. For instance, the Bangla (108.5M) model achieves lower perplexity (4.102) than the larger Hindi (367.5M) model (11.052), suggesting that corpus quality, domain diversity, and language-specific characteristics can dominate scaling effects in certain settings (Conneau et al., 2020a).

In terms of efficiency, CPU inference speed decreases substantially with model size (e.g., 37.35 tokens/sec for Bangla 108.5M vs. 12.91 tokens/sec for Hindi 367.5M), reflecting the expected computational trade-off in transformer-based architectures (Pope et al., 2022). Models in the 100M–200M range (e.g., mParamanu 162M, Marathi 207.73M) provide a more favorable balance between perplexity and latency, making them practical for CPU-bound deployment.

MFU remains consistently high (∼39–40%) for most models, indicating efficient utilization of compute during training (Chowdhery et al., 2024). However, lower MFU observed for Marathi (19.50%) and Tamil (18.77%) suggests potential inefficiencies in batching or input pipeline design, which may limit achievable throughput. Overall, these results indicate that scaling alone does not guarantee uniform gains across languages, and that both dataset characteristics and system-level efficiency are critical for achieving optimal performance.

N-shot	XNLI-Hindi	XStoryCloze-Hindi	XStoryCloze-Telugu	XCOPA-Tamil
0	33.49	**52.42**	**56.06**	**54.00**
5	**34.04**	51.49	54.67	52.40
25	33.23	52.02	55.92	49.80

Table 13: N-shot evaluation of pretrained Paramanu models across various benchmarks.

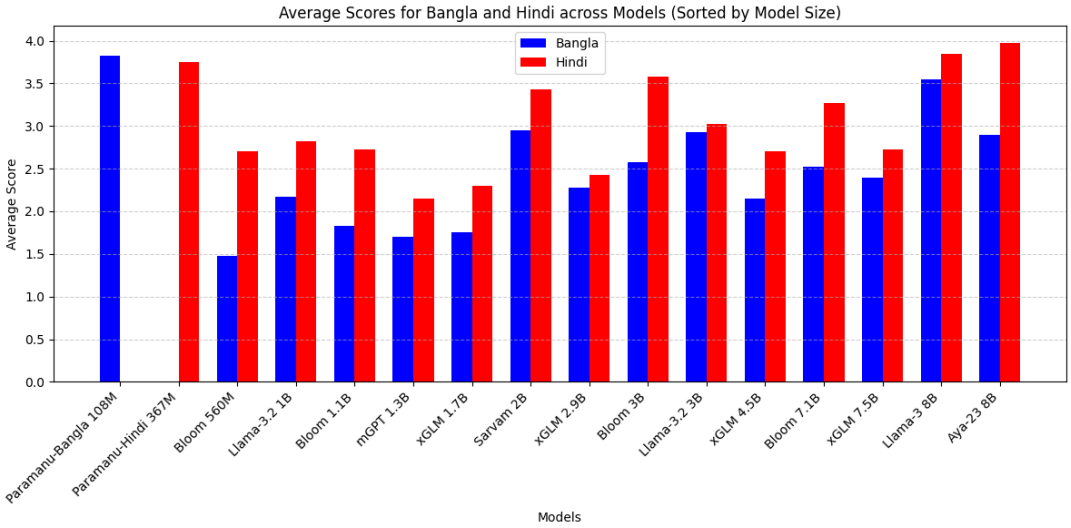


Figure 3: Human evaluation of LLMs. Average score across Grammar, Coherence, Creativity, Factuality.

Model	Perplexity	MFU	CPU Inference Speed
Bangla 108.5M	4.102	22.57	37.351
Hindi 367.5M	11.052	39.87	12.906
mParamanu 162M	6.924	39.93	34.711
Marathi 207.73M	8.943	19.50	24.875
Tamil 207.84M	7.618	18.77	24.535
Telugu 208.25M	5.400	40.07	24.125

Table 14: CPU Inference speed (tokens/sec, FP32), perplexity of our models and MFU metric during training. MFU is Model FLOPs Utilization metric.

D Model & Training Configuration

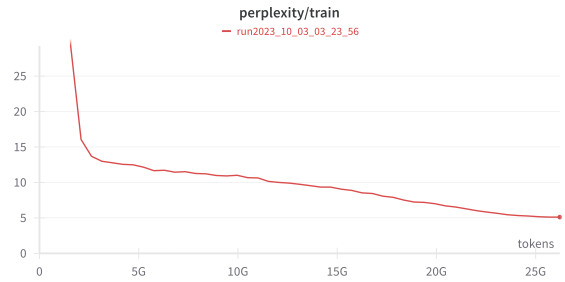
In this section, we provide details on the various model architecture and pretraining hyperparameters used in our experiments. Tables 15 and 16 list these configurations. Figure 4 presents plots of training perplexity versus training tokens for several of our pretrained models. The results indicate that training perplexity decreases and converges as the number of training steps and tokens increases.

n_params	d_model	n_layers	n_heads	n_kv_heads	Seq Len
108M	768	12	12	12	1024
139M	896	14	14	14	1024
162M	1024	12	16	16	1024
208M	1024	16	16	16	1024
237M	1024	18	18	18	1024
367M	1280	18	10	10	1024

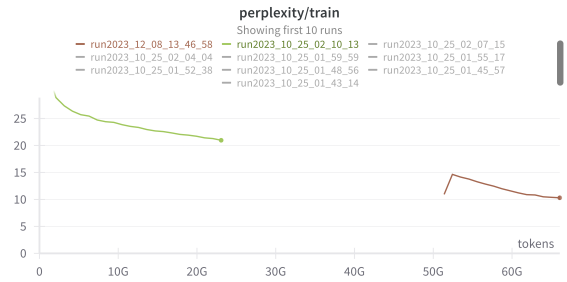
Table 15: Model size configuration

Model	Params	Batch	Grad Acc.	Seq Len	LR
Bangla	108.5M	32	8	1024	0.003
Hindi	162M	32	8	1024	0.002
Hindi	367.5M	32	16	1024	0.003
Marathi	207.73M	32	8	1024	0.003
mParamanu	92.63M	32	16	1024	0.002
mParamanu	162M	32	8	1024	0.002
Sanskrit	139.3M	64	8	1024	0.003
Tamil	207.84M	32	8	1024	0.002
Telugu	208.25M	16	16	1024	0.003

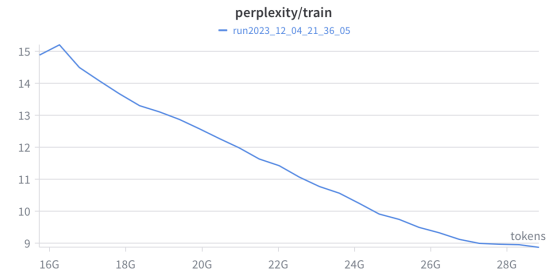
Table 16: Training hyperparameters for various Paramanu models. All models are pretrained for 100K training steps except Hindi 367M (150K).



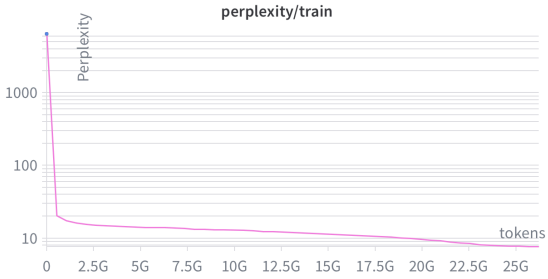
(a) Training perplexity of Paramanu-Bangla vs. tokens (in billions). Each color indicates a resumed run.



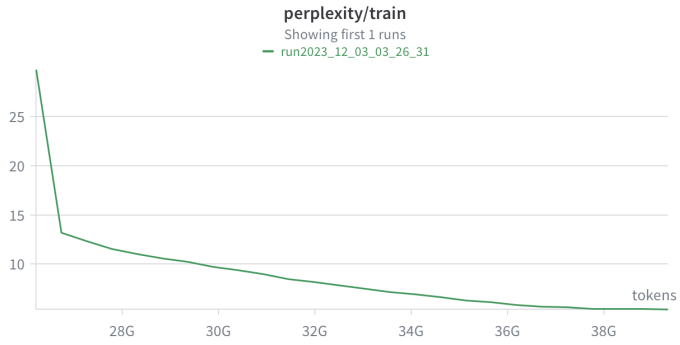
(b) Training perplexity of Paramanu-Hindi vs. tokens (in billions). Each color indicates a resumed run.



(c) Training perplexity of Paramanu-Marathi vs. tokens (in billions). Each color indicates a resumed run.



(d) Training perplexity of Paramanu-Tamil vs. tokens (in billions). Each color indicates a resumed run.



(e) Training perplexity of Paramanu-Telugu vs. tokens (in billions). Each color indicates a resumed run.

Figure 4: Training perplexity trends for Paramanu pretrained models, shown against tokens. Colors denote runs resumed after interruptions.