Instruction-Tuned Urdu LLMs: Efficient Adaptation of Llama Models and Evaluation Resources for Urdu

Munief Hassan Tahir, Sana Shams, Sarmad Hussain, Miriam Butt
Centre for Language Engineering , Al-Khawarizmi Institute of Computer Science
, University of Engineering and Technology, Lahore, Pakistan
University of Konstanz, Germany
{munief.tahir, sana.shams, sarmad.hussain}@kics.edu.pk
miriam.butt@uni-konstanz.de

Abstract

This paper presents UrduLLaMA 1.1 and UrduLLaMA 1.1 Tiny, two instruction-tuned large language models (LLMs) designed to advance natural language processing for Urdu, a low-resource language with limited representation in multilingual corpora. These instruction-tuned models are derived from Llama-3.1-8B-Instruct and Llama-3.2-3B-Instruct architectures, respectively by conducting continual pretraining on 800 million diverse Urdu tokens curated from public and proprietary sources, followed by Supervised Fine-Tuning (SFT) using LoRA on 432K Urdu instructions spanning diverse NLP tasks. Rigorous evaluation across 14 culturally-specific domains using our novel Urdu LLM Evaluation Dataset demonstrates superior performance. UrduLLaMA 1.1 achieves 65.3 average accuracy (GPT-5 Nano evaluation), outperforming its Llama-3.1-8B-Instruct base (50.7) across all categories and surpassing Llama-3.3-70B-Instruct (62.7) in 8 out of 14 domains. UrduLLaMA 1.1 Tiny transforms Llama-3.2-3B-Instruct (38.8) into a (61.2) performer. Human evaluation by native Urdu linguists confirms these gains (3.51/5 vs. 2.61/5 base). Our results validate targeted adaptation strategies combining continual pretraining with instruction tuning as computationally efficient solutions for low-resource languages, enabling state-of-the-art Urdu LLM models with accessible hardware.

Keywords: Urdu LLM, Low-Resource Language, Continual Pretraining, Instruction Tuning, Llama-3, LoRA, SFT, Multilingual Adaptation

1. Introduction

Language modeling has undergone profound transformation, propelled by the swift advancement of Large Language Models (LLMs) that have redefined benchmarks for natural language understanding and generation. Proprietary models like OpenAI's ChatGPT (OpenAI, 2022) demonstrate remarkable capabilities but are limited by their closed nature, which hinders research accessibility. In contrast, open models like LLaMA (Grattafiori et al., 2024) and Mistral (Jiang et al., 2023), despite their smaller scale, deliver competitive performance across multiple languages. However, multilingual LLMs face challenges in handling low-resource languages such as Urdu, largely due to their limited representation in training datasets. This data gap leads to restricted vocabulary coverage and inadequate encoding representations, ultimately reducing model performance on Urdu NLP tasks, as shown in recent benchmark studies (Arif et al., 2024; Tahir et al., 2025). Overcoming this limitation is crucial to unlocking the full potential of LLMs for Urdu and other underrepresented languages. In this research, we tackle this challenge by developing Urdu-specific LLMs. We begin by continually pretraining Llama-3.1-8B-Instruct (Grattafiori et al., 2024) and Llama-3.2-3B-Instruct on 800 million Urdu tokens to enhance the model's foundational representation of the language. This is followed by instruction fine-tuning on 432K Urdu-centric instructions to further improve their alignment and conversational capability.

To evaluate model performance, we conduct extensive assessments across 14 distinct domains using a curated dataset designed to captures local language proficiency and domain-specific understanding. For comprehensive evaluation, both human assessments and GPT-based judgments are employed on this dataset to ensure the reliability and consistency of results. The model outcomes are compared against the corresponding base models and *Alif* (Shafique et al., 2025), an open-source Urdu language model. Additionally, evaluations on translated benchmark datasets are performed to provide further comparative insights.

The paper is structured as follows: Section 1 introduces the study and Section 2 presents the related work. Section 3 details the dataset curation and Section 4 explains the steps taken to preprocess the data. This is followed by Section 5, which describes the development process of UrduLLaMA 1.1 and UrduLLaMA 1.1 Tiny models including the experimental and training details, and Section 6, which covers the evaluation and discussion leading to Section 7, which concludes the paper.

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 10603–10616
11–16 May 2026.

2. Related Work

Apart from *Alif* (Shafique et al., 2025), a model based on Llama-3.1-8B-Instruct by continually pretraining it on 200K Urdu Wikipedia articles and fine-tuned on 105K instruction, — no other known study has explored continual pretraining of LLMs for Urdu. This section reviews the most relevant related efforts, including models developed for Asian and other low-resource languages built using the LLaMA framework.

Tamil-Llama (Balachandran, 2023), an Asian language model built on LLaMA 2 (Touvron et al., 2023), incorporates 16,000 Tamil tokens and utilizes the Low-Rank Adaptation (LoRA) (Hu et al., 2021) technique for efficient training on Tamil datasets. The model was trained on an Nvidia A100 GPU with 80GB of VRAM for 48 hours, followed by instruct fine tuning on translated Alpaca datasets (Taori et al., 2023) and a custom subset of the OpenOrca (Lian et al., 2023) dataset for 60 hours using Microsoft Azure's Standard NC24 ads A100v4 instance. Performance evaluations indicate substantial improvements in Tamil text generation, with the *Tamil-Llama 13B* model outperforming OpenAI's GPT-3.5-turbo on Tamil language tasks.

Taiwan-LLM (Lin and Chen, 2023), an LLM for Traditional Chinese, underwent continual pretraining on LLaMA-2 using 35.1 billion tokens and a diverse instruction set derived from 17 fine tuning datasets, including 20,000 user feedback instances. The training process leveraged the Transformer Reinforcement Learning (TRL) library (Hu et al., 2023), along with DeepSpeed ZeRO-2 (Rajbhandari et al., 2020) and FlashAttention-2 (Dao, 2023) to optimize memory usage and enhance training efficiency. Utilizing up to 48 NVIDIA H100 Tensor Core GPUs, *Taiwan-LLM* demonstrated superior performance in understanding and generating Traditional Chinese text, surpassing models such as GPT-4 and Claude-2.1 (Anthropic, 2023).

PersianLLaMA (Abbasi et al., 2023), the first large-scale Persian language model, was trained from scratch on 184 million tokens from Persian Wikipedia and 9 billion tokens from the OSCAR dataset (Suárez et al., 2020). The training process leveraged DeepSpeed (Rasley et al., 2020) and TencentPretrain (Zhao et al., 2023), two advanced frameworks for optimizing deep learning, utilizing two A100 GPUs with 80GB of VRAM over 12 days. Additionally, they conducted an experiment using LoRA with the original English LLaMA weights, training on a single A100 GPU with 80GB of VRAM for over 70 hours. Their evaluations indicate that *PersianLLaMA* significantly outperformed its competitors in both understanding and generating Persian text.

AceGPT (Huang et al., 2023) is an Arabic-centric large language model developed on top of LLaMA 2 (Touvron et al., 2023). It was localized through pretraining on extensive Arabic corpora, followed by Supervised Fine-Tuning (SFT) with native Arabic instructions and Reinforcement Learning with AI Feedback (RLAIF) using culturally aligned preference data. The training setup employed 24 NVIDIA A100 80GB GPUs, with a 2048-token context length, AdamW optimizer, cosine learning rate scheduler, and gradient accumulation of 128. Evaluation results show that *AceGPT* achieved state-of-the-art performance across several benchmarks, including Arabic Vicuna-80, AlpacaEval, ALUE, MMLU, EXAMs, and the Arabic Cultural and Value Alignment (ACVA) dataset, establishing it as a leading open-source Arabic LLM. Despite its strong performance, *AceGPT* faces challenges such as a vocabulary largely restricted to Arabic letters, limiting encoding efficiency, and pretraining constrained by machine resources.

Airavata (Gala et al., 2024) is an instruction-tuned model for Hindi, built by fine tuning *OpenHathi* (SarvamAI, 2023), on 404k instruction instances from diverse Hindi instruction-tuning datasets. *OpenHathi* (SarvamAI, 2023) is again a model built on the LLaMA 2 7B architecture. The training employed both full fine tuning and supervised fine tuning using LoRA. Their results demonstrated that *Airavata* significantly outperforms *OpenHathi* on most tasks, highlighting the effectiveness of fine tuning in aligning the base model to a variety of tasks.

SeaLLMs (Nguyen et al., 2024b) is an innovative series of language models focused on Southeast Asian (SEA) languages. Built upon LLaMA 2 (Touvron et al., 2023) and Mistral 7B (Jiang et al., 2023), SeaLLMs underwent continued pretraining with an extended vocabulary, followed by a hybrid approach for instruction and alignment tuning. Their evaluation claims that *SeaLLMs* significantly outperform ChatGPT-3.5 in non-Latin languages, such as Thai, Khmer, Lao, and Burmese, by large margins, while remaining lightweight and cost-effective to operate.

A related study on *Chinese LLaMA*(Cui et al., 2023) extended different variants of LLaMA 2 (Touvron et al., 2023) by adding 20,000 Chinese tokens to the existing vocabulary. The model was pre-trained using LoRA and fine tuned on Chinese instruction datasets formatted according to Alpaca (Taori et al., 2023). Training was conducted on A40 GPUs (48GB VRAM), with up to 48 GPUs used depending on the model size. The parameter-efficient training with LoRA was carried out using the PEFT library[1]. Experimental results demonstrate significant improvements in Chinese text understanding and generation over the original LLaMA.

[1] https://github.com/huggingface/peft

Source	Original Token Count	Token Count After Processing	Reduction	Percentage Reduction (%)
Publically Available Resources	798,260,573	541,151,638	257,108,935	32.20
Inhouse	639,786,525	606,053,446	33,733,079	5.30
Dataset	1,438,047,098	1,147,205,084	290,842,014	-
Dataset (in Billion)	**1.43**	**1.14**	**0.29**	-

Table 1: Summary of Token Count Reduction Across Different Data Sources)

Another study (Chen et al., 2024b) conducted a two-stage continual pretraining of Llama-3-8B (Grattafiori et al., 2024) for Chinese. Initial experiments were conducted on *TinyLLaMA* (Zhang et al., 2024), followed by training on 100 billion tokens and fine-tuning on synthetic scientific QA data. Results across multiple benchmarks show substantial performance gains in both general and scientific reasoning abilities without degrading the backbone model's original capacities.

VinaLLaMA (Nguyen et al., 2023), an open-weight, state-of-the-art (SOTA) Vietnamese LLM, was built upon LLaMA 2 (Touvron et al., 2023) with an additional 800 billion trained tokens followed by fine tuning on 1 million sample instruction of Vietnamese and English. They claim to achieve state-of-the-art results on different key benchmarks, showcasing fluency in Vietnamese and a deep understanding of their culture.

In summary, this literature review highlights the absence of dedicated large-scale Urdu LLMs and underscores the pressing need for such resources. To address this gap, this paper presents the first large-scale continual pretraining and fine-tuning of Llama-3.1-8B-Instruct and Llama-3.2-3B-Instruct for Urdu, leveraging 800 million Urdu tokens and applying LoRA-based instruction tuning. This work represents a pioneering step toward developing advanced, efficient LLMs for Urdu and other underrepresented languages.

3. Dataset Curation

A pivitol challenge for building LLMs, particularly in low resource languages, is the availability of sizeable high-quality data for building foundation LLMs. As the quality and diversity of data significantly influence the capabilities of LLMs (Chen et al., 2024a), we supplemented our in-house dataset with data from several publicly available sources, including CC-100 (Wenzek et al., 2020), the Urdu corpus from OSCAR (Suárez et al., 2020), the Urdu Web Corpus (Shafiq et al., 2020), Urdu data from XL-Sum (Hasan et al., 2021), and (Goldhahn et al., 2012). The raw text underwent a comprehensive pre-processing pipeline, outlined in Section 4, to ensure language-specific content, maintain quality, and remove duplicates. A summary of the collected datasets and the impact of processing is provided in Table 1.

4. Preprocessing Pipeline

This section outlines the pre-processing steps applied to construct our dataset. We primarily adopted approaches similar to those proposed by (Zeng et al., 2021), (Ennen et al., 2023), and (Lu et al., 2024). The steps are as follows:

- **Language Filtering:**

  This step was performed at the document level to retain only language-rich documents. Similar approaches have been adopted by others, such as the Falcon team (Penedo et al., 2023) for creating RefinedWeb, where they used the fastText language classifier from CC-Net (Wenzek et al., 2020) at the document level. This method has also been utilized by (Nguyen et al., 2024a) for building CulturaX and by (Laurençon et al., 2022) for constructing the BigScience ROOTS corpus. In our case, we conducted language filtering using the CLE Urdu Language Identification API[2], applying a threshold of 0.9 to ensure the retention of predominantly Urdu documents. To validate our choice of the CLE Urdu Language Identification API, we conducted experiments comparing it with fastText, assessing the effectiveness of both in identifying Urdu content within documents containing varying proportions of Urdu and non-Urdu text. The results of which are summarized in Table 2, demonstrating that the CLE Urdu Language Identification API provided scores more aligned with the expected composition of test data.

- **Data Standardization:** Data standardization involves the normalization and transformation of text data to make it more manageable and comprehensible during the model training process (Lu et al., 2024). Since syntax of the Urdu language requires specialized techniques, we applied as described in (Nazir et al., 2024). Major steps include Unicode-based filtering, replacing non-standard characters with their standard forms, and handling Urdu-specific features such as end symbols, poetic symbols, and quotation marks. Additionally, some documents had varying lengths, so we split the text to maintain an average context length of 512 tokens.

[2] https://tech.cle.org.pk/api_langid

File Composition	CLE Urdu Language Identification API	fastText Model Score
80% Urdu, 20% non-urdu	0.827	lang: ur , prob: 0.991
50% Urdu, 50% non-urdu	0.503	lang: ur , prob: 0.847
25% Urdu, 75% non-urdu	0.257	lang: en , prob: 0.439
100% Urdu	1.000	lang: ur , prob: 0.997
100% Urdu (with urdu numerials)	1.000	lang: ur , prob: 0.994
Only Numericals	0.000	lang: ru , prob: 0.349

Note: The "lang" represents the detected language, "prob" represents the probability score. The language codes are as follows: "ur" = Urdu, "en" = English, "ru" = Russian.

Table 2: Language Identification Experiments with CLE Urdu Language Identification API and fastText

- **Quality Filtering:** To enhance the dataset's quality, motivated by the data processing pipeline from (Laurençon et al., 2022) and (Nguyen et al., 2024a), we utilized various dataset metrics to identify and filter outlying documents. Filtering was applied based on stopword ratios, flagged word ratios, and empty documents. The threshold values for the stopword ratio and flagged word ratio were set at 0.1 and 0.025, respectively, which align with the threshold values used in the BigScience ROOTS project (Laurençon et al., 2022).

  In addition to filtering, we implemented Personally Identifiable Information (PII) removal to protect sensitive data. We employed rule-based approach leveraging regular expressions regexes library to detect and remove sensitive information such as phone numbers, identification numbers, and email addresses. These measures ensured that the dataset was free from personally identifiable information, enhancing privacy and usability for model training.

- **Deduplication:**

  Despite thorough data cleaning, the remaining, dataset still contain a substantial amount of repeated data due to various reasons, including information being reposted on the web, multiple references to the same articles and plagiarism. The duplicated data can thus cause memorization and significantly hinder generalization for LLMs (Lee et al., 2022). Therefore deduplication is required as it decreases memorization of training data (Kandpal et al., 2022). Initially, deduplication was performed within individual datasets, followed by an overall deduplication across all datasets to address potential similarities among different sources. We applied deduplication at two levels with Table 3 summarizes the results of this process:

Step	Token Count
Original Dataset	1.43 Billion
After Preprocessing	1.14 Billion
After Overall Deduplication	1.08 Billion

Table 3: Impact of Deduplication on Dataset

1. **Exact Document Deduplication:** We applied the SimHash technique, as used in the creation of WuDaoCorpora (Yuan et al., 2021) corpus, Roots Corpus for BigScience's BLOOMZ model (Abadji et al., 2022) and (Laurençon et al., 2022), to perform deduplication. A hash was generated from the content of each document (ignoring spaces) to uniquely identify it. If a duplicate hash was found, the corresponding document was removed.
2. **Inside Document Deduplication:**

   The second step involved deduplicating individual lines within the documents. Following the approach outlined by (Laurençon et al., 2022), we performed a line-by-line comparison to identify and remove repeated content. Duplicate lines, regardless of their position within a document, were eliminated.

5. UrduLLaMA 1.1 Models

Llama-3.1-8B-Instruct, as introduced in (Grattafiori et al., 2024) by Meta, is built upon an extensive pretraining corpus of 15 trillion tokens, while Llama-3.2-3B, used for UrduLLaMA 1.1 Tiny, leverages up to 9 trillion tokens. We utilize these model architectures for continual pretraining due to their open-source availability and inclusion of Urdu language data, making them suitable for our research. The complete process of creating UrduLLaMA 1.1 and UrduLLaMA 1.1 Tiny, illustrated in Figure 1, follows four key stages: data collection, data processing,

continual pretraining, and fine-tuning, each enhancing the models' linguistic understanding and task adaptability.

5.1. Continual Pretraining

The UrduLLaMA 1.1 model is trained on the Causal Language Modeling (CLM) task, enabling it to predict and generate the next word in a sequence. This stage plays a crucial role in refining LLaMA's proficiency in Urdu by allowing the model to grasp the language's intricate syntactic structures, semantic nuances, and unique linguistic traits. Leveraging its autoregressive nature, CLM mirrors the human process of language comprehension and generation, which is inherently context-dependent. Consequently, by the end of this initial training phase, LLaMA acquires the ability to generate and interpret Urdu text with contextual relevance and linguistic accuracy.

5.1.1. Pretraining Dataset

Due to hardware limitations, about 800 million tokens were selected for continual pretraining of Llama-3.1-8B-Instruct and Llama-3.2-3B-Instruct models from the curated dataset as explained in Section 3.

5.1.2. Pretraining Setup

The foundational model of UrduLLaMA 1.1 is initialized with the original Llama-3.1-8B-Instruct weights and UrduLLaMA 1.1 Tiny is intitialized with the original Llama-3.2-3B-Instruct weights and underwent pretraining using the *bf16* precision setting. Our pretraining strategy involved parameter-efficient finetuning with Low-Rank Adaptation (LoRA), where LoRA adapters were applied to attention modules (q_proj, v_proj, and output_proj) and the MLP layers, with a LoRA rank of 64, alpha of 128, and dropout of 0.1. The training utilized the torchtune library (PyTorch Contributors, 2024) for distributed LoRA finetuning. An AdamW optimizer was employed with a learning rate of 2e-4, weight decay of 0.01, and a cosine learning rate scheduler with 100 warmup steps. Gradient accumulation steps of 8 were used with a batch size of 1, resulting in an effective batch size of 8. Memory optimization techniques such as activation checkpointing, activation offloading, and PyTorch compilation were applied to manage the large model size effectively. Pretraining was conducted on 3 NVIDIA L40 48GB GPUs, with the training process spanning approximately 892 hours for UrduLLaMA 1.1 model and 530 hours for UrduLLaMA 1.1 Tiny. Both the models were trained for 1 epoch on the dataset.

5.2. Instruct Tuning

Language models pre-trained using the causal language modeling (CLM) objective often struggle to follow user instructions and sometimes generate irrelevant or unintended content (Balachandran, 2023). This limitation arises because the CLM objective is designed to predict the next token in a sequence rather than understand or respond to instructions effectively (Ouyang et al., 2022). To address this issue and align the model's behavior with user intentions, we employed instruction fine-tuning using Supervised Fine-Tuning (SFT). This step refines the LLM's capabilities, allowing it to interpret and execute task-specific instructions more effectively in natural language. Instruction fine-tuning focuses on a wide array of tasks articulated through language, ensuring the LLM's adaptability without task-specific alterations (Wei et al., 2022).

5.2.1. Instruct Tuning Datasets

To enhance the model's ability to follow Urdu-specific instructions, we curated a comprehensive dataset for SFT, totaling 432,759 unique instruction-response pairs after merging and deduplication. The dataset was derived from two main categories: translated datasets from established English instruction-tuning resources and generated datasets from Urdu news articles. These cover diverse NLP tasks such as question answering, reasoning, summarization, classification, and creative writing. Table 4 summarizes the composition and statistics of these datasets.

Category	Deduplicated Rows
Translated Datasets	259,886
Generated Datasets	172,873
Total Combined	432,759

Table 4: Summary of Instruct Tuning Datasets

Translated Datasets. These were derived from high-quality English instruction-tuning datasets, translated into Urdu to adapt them for our model. The Urdu Alpaca (26,019 rows) and Urdu Dolly (15,015 rows) datasets were sourced directly from existing repositories (Khalil, 2024; Saeed, 2023), based on the Stanford Alpaca (Taori et al., 2023) and Dolly (Conover et al., 2023) datasets, respectively. The Urdu Self-Instruct dataset is a curated translation of the Self-Instruct dataset (Wang et al., 2023), containing 52,000 high-quality instruction-output pairs covering tasks like reasoning, summarization, and transformation. The Urdu FLAN, comprising approximately 140,000 instances, was translated from the FLAN collection (Chung et al., 2022), which emphasizes instruction-following across mul-

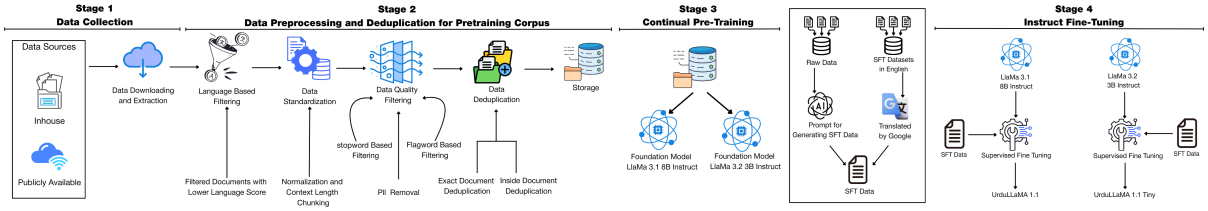

Figure 1: Development of UrduLLaMA 1.1 and UrduLLaMA 1.1 Tiny

tiple NLP benchmarks. Translations for Urdu Self-Instruct and Urdu FLAN were performed using the Google Translate API, followed by manual spot-checking to ensure quality. After merging all translated sources (268,840 rows), we applied deduplication using exact string matching on instruction-input-output triples, removing 8,954 duplicates to yield 259,886 unique rows.

Generated Datasets. To incorporate culturally relevant and domain-specific Urdu content, we collected high quality Urdu content which included articles related to politics, entertainment, sports, and general news. This content was used to generate instruction-based data in a semi-supervised manner via the GPT-4o-mini API (OpenAI, 2024). For each article, we provided the content along with a custom prompt instructing the model to generate at least five diverse, self-contained Alpaca-style (Taori et al., 2023) instruction-input-output pairs in Urdu, covering tasks like factual questioning, summarization, or scenario analysis. The prompt, inspired by (Wang et al., 2023), ensured variety and independence from external context.

This process resulted in 172,877 rows, with minor deduplication reducing it to 172,873 unique rows. The generated data enhances the model's ability to handle real-world Urdu contexts, mitigating biases from translated sources. The final merged dataset (432,759 rows) was shuffled to ensure a balanced distribution of translated and generated instances for training.

5.2.2. Instruct Tuning Setup

For instruction fine-tuning, we utilized the Low-Rank Adaptation (LoRA) method, integrating LoRA adapters into the attention layers (q_proj, v_proj, output_proj) and MLP layers of the Llama-3.1-8B-Instruct model for UrduLLaMA 1.1 and Llama-3.2-3B-Instruct model for UrduLLaMA 1.1 Tiny. The embeddings, language model head, and LoRA parameters were trained using the torchtune library (PyTorch Contributors, 2024) for efficient distributed training. The training for UrduLLaMA 1.1 Tiny was conducted on a single NVIDIA A100 40GB GPU, with a batch size of 1, FP16 precision, and a maximum sequence length of 512. We used an AdamW optimizer with an initial learning rate of 2e-4, a dropout rate of 0.1, a LoRA rank of 64, and a LoRA alpha of 128. The model was trained for 3 epochs, with a total training time of approximately 72 hours. The training for UrduLLaMA 1.1 was conducted on 3 NVIDIA L40 48 GB GPUs for one epoch with a total training time of approximately 190 hours.

6. Evaluation

6.1. Urdu LLM Evaluation Dataset

To address the absence of a dedicated Urdu evaluation dataset, we developed a novel benchmark inspired by methodologies from prior language-specific LLaMA adaptations (Balachandran, 2023; Abbasi et al., 2023; Cui et al., 2023). The dataset, manually curated by linguists to reflect Urdu speaking contexts, comprises 350 instances across 14 diverse categories (e.g., business, literature, health, local law), with 25 instances per category. These categories encompass tasks like reasoning, summarization and text generation, ensuring comprehensive evaluation of model capabilities.

6.2. GPT as Judge Evaluation

Using the Urdu LLM Evaluation Dataset 6.1, we employed GPT-5 Nano (OpenAI, 2025) as an impartial judge to evaluate model responses. For each question-response pair, with model identities concealed, the judge rated responses on five criteria: clarity, coherence, completeness, relevance, and accuracy. Each criterion was scored on a 1 to 5 scale (1 = poor, 5 = excellent). For each category (e.g., business), we aggregated scores across the 25 instances by summing the ratings for each criterion separately and then computing the total sum across all five criteria. The category-level average score was obtained by dividing this total by 625 (the maximum possible sum: $5 \times 25 \times 5$). Table 5 presents these average scores per category for the evaluated models: Llama 3.1 8B Instruct, Llama 3.2 3B Instruct, Llama 3.3 70B Instruct, Alif, UrduLLaMA 1.1 Tiny, and Urdu Instruct.

UrduLLaMA 1.1 achieved the highest average rating of 65.3, surpassing its base Llama 3.1 8B model (50.7) in all 14 categories, with standout

Category	Llama 3.2 3B	Llama 3.1 8B	UrduLLaMA 1.1 Tiny	Llama 3.3 70B	Alif	Urdu Instruct
Business	30.2	38.2	65.0	46.7	59.4	64.0
Close-ended QA	45.0	75.2	64.3	85.8	78.6	71.7
Common sense QA	45.8	57.9	60.8	78.2	64.2	67.2
Literature	40.3	52.5	65.8	59.5	68.6	63.2
Political	41.9	48.2	67.5	61.1	72.8	73.8
Reasoning	48.5	65.3	61.8	87.2	47.4	66.4
Sports	36.0	42.9	53.3	59.7	55.8	54.1
Story writing	28.0	31.0	52.8	35.4	68.0	65.3
Summarization	71.4	81.4	73.8	88.3	76.3	76.5
Agriculture	29.9	32.8	48.0	51.2	52.5	59.0
Banking	25.8	40.5	54.2	51.4	57.6	59.4
Health	32.0	46.9	66.1	57.4	66.1	68.2
Local law	32.8	50.7	56.3	58.9	56.6	57.1
Urdu news	35.8	46.7	66.6	57.3	68.6	68.0
Average	38.8	50.7	61.2	62.7	63.7	65.3

Table 5: GPT as a Judge Evaluation Results (Acc. %)

performances in political discourse (73.8 vs. 48.2) and health (68.2 vs. 46.9). It also outperformed the larger Llama 3.3 70B (62.7) in 11 categories, including literature (63.2 vs. 59.5) and Urdu news (68.0 vs. 57.3).

UrduLLaMA 1.1 Tiny, built on the smaller Llama 3.2 3B (38.8), attained a 61.2 average accuracy, exceeding its base model in all 14 categories, with notable gains in business (65.0 vs. 30.2) and story writing (52.8 vs. 28.0), and outpacing the 70B model in 7 categories, such as political (67.5 vs. 61.1) and literature (65.8 vs. 59.5).

These results demonstrate the models' enhanced ability to handle culturally nuanced and domain-specific Urdu tasks, leveraging targeted adaptations to deliver precise, contextually relevant outputs. The robust performance of UrduLLaMA 1.1 Tiny model demonstrates its effectiveness in low-resource settings. Its ability to operate efficiently on resource-limited devices without compromising accuracy broadens the accessibility of advanced Urdu language processing for real-world applications in domains such as education, healthcare, and local governance.

6.3. Human Evaluation

To complement the automated evaluation in Section 6.2 and validate model performance with culturally nuanced insights, we conducted a human evaluation using the Urdu LLM Evaluation Dataset described in Section 6.1. A panel of three native Urdu speaking evaluators, working independently to ensure unbiased ratings, assessed anonymized responses from six models across all 350 instances spanning 14 categories. Each response was rated on a scale of 1 to 5, with 5 representing the highest satisfaction and understanding, and 1 the lowest. For each category and model, individual evaluator ratings were first averaged to obtain a per-category score per model. These category-level averages were then averaged across all 14 categories to compute the final average score for each model. This process yielded the final average scores across all categories and evaluators, which were used to rank the models as presented in Table 6.

Model	Average Rating
Llama 3.2 3B	2.06
Llama 3.1 8B	2.61
Alif	3.08
UrduLLaMA 1.1 Tiny	3.17
Llama 3.3 70B	3.46
UrduLLaMA 1.1	3.51

Table 6: Summary of Human Evaluation Results

The human evaluation results corroborate the trends observed in the GPT-5 Nano assessment, with UrduLLaMA 1.1 achieving the highest average rating of 3.51, closely followed by Llama 3.3 70B at 3.46, while significantly outperforming its base Llama 3.1 8B (2.61). UrduLLaMA 1.1 Tiny also demonstrated strong performance at 3.17, surpassing Alif (3.08) and its base Llama 3.2 3B (2.06). These outcomes align with the models' excellence in culturally attuned tasks highlighted in the GPT evaluation, such as political discourse and health related queries. The human evaluators as native Urdu speakers and local linguists, captured subtle cultural nuances, idiomatic expressions, and contextual relevancies that automated judges like GPT-5 Nano might overlook, providing a more grounded validation of the models' effectiveness in real world, domain-specific Urdu applications. This human centeric approach complements the benchmark evaluations by emphasizing practical utility in low resource settings, further underscoring the value of targeted Urdu adaptations for accessible AI deployment.

6.4. Evaluation on Benchmark Datasets

To evaluate UrduLLaMA 1.1's capabilities in reasoning and knowledge-intensive tasks, we developed Urdu-translated versions of standard benchmarks using Google Translate. These benchmarks include the Massive Multitask Language Understanding (MMLU) dataset (Hendrycks et al., 2021), HellaSwag (Zellers et al., 2019), Multilingual Grade School Math (MGSM) (Shi et al., 2023), and the AI2 Reasoning Challenge (ARC) datasets, comprising ARC Easy and ARC Challenge (Clark et al., 2018).

Results, shown in Table 7 for MMLU and HellaSwag, and Table 8 for MGSM, ARC Easy, and ARC Challenge, assess UrduLLaMA 1.1's performance in commonsense reasoning, scientific deduction, and mathematical problem-solving.

UrduLLaMA 1.1 excels in structured scientific reasoning (ARC) but lags behind its base model on knowledge-intensive tasks, revealing complementary strengths. On MMLU and Hellaswag (Table 7), base Llama 3.1 8B outperforms UrduLLaMA 1.1. MMLU requires encyclopedic recall across 57 subjects, leveraging the base model's broader multilingual pre-training. Hellaswag demands commonsense inference in narrative contexts, where the base excels due to its extensive exposure to diverse patterns. UrduLLaMA 1.1's instruction tuning prioritizes targeted reasoning over expansive knowledge assimilation, making it less suited for these breadth-oriented tasks.

Model	Hellaswag	MMLU	Average
Alif	12.11	29.80	20.96
Llama 3.2 3B	27.22	20.39	23.81
Llama 3.1 8B	32.01	31.64	31.83
UrduLLaMA 1.1	27.45	25.63	26.54

Table 7: Translated Hellaswag & MMLU Results

When evaluated on scientific reasoning benchmarks, UrduLLaMA 1.1 leads on ARC Easy (41.37) and ARC Challenge (29.69) (Table 8), surpassing all baselines. These scientific reasoning benchmarks emphasize step-by-step logical deduction—from basic inferences (Easy) to adversarial puzzles (Challenge)—which align perfectly with our Urdu-specific instruction tuning. However, on MGSM (6.4 vs. base's 12.0), the base model's stronger multilingual mathematical grounding provides an edge in word problem solving. UrduLLaMA 1.1 demonstrates that Urdu instruction tuning boosts logical reasoning (ARC success) while the base Llama retains advantages in knowledge breadth (MMLU, Hellaswag).

This highlights UrduLLaMA 1.1 as a specialized reasoning model, with opportunities to combine both strengths through expanded Urdu knowledge training for comprehensive performance.

Model	MGSM	Arc E.	Arc C.	Average
Alif	5.2	15.61	11.52	10.78
Llama 3.2 3B	3.6	23.4	23.63	16.88
Llama 3.1 8B	12.0	38.01	23.89	24.63
UrduLLaMA 1.1	6.4	41.37	29.69	25.82

Table 8: Translated MGSM & ARC Results

7. Conclusion

This work presents UrduLLaMA 1.1 and UrduLLaMA 1.1 Tiny, specialized instruction-tuned models that significantly advance Urdu language processing capabilities. Through continual pretraining on 800 million diverse Urdu tokens followed by SFT on 432K instructions, our models transform commodity multilingual architectures into Urdu specialists. The novel Urdu LLM Evaluation Dataset, spanning 14 categories, enabled rigorous assessment. GPT-5 Nano evaluation revealed Urdu Instruct's dominance (65.3 average) over its base Llama 3.1 8B (50.7) across all categories and even surpassing the bigger variant Llama 3.3 70B (62.7) in 8/14 domains. UrduLLaMA 1.1 Tiny similarly transformed its base Llama 3.2 3B (38.8) into a formidable 61.2 performer. Human evaluation by native Urdu linguists validated these findings, with UrduLLaMA 1.1 (3.51) and Llama 3.3 70B (3.46) leading, while UrduLLaMA 1.1 Tiny (3.17) substantially outperformed baselines and the Alif model. Benchmark results highlight complementary strengths: UrduLLaMA 1.1 excels in scientific reasoning (ARC Easy: 41.37, ARC Challenge: 29.69) while base models retain advantages in knowledge-intensive tasks (MMLU, Hellaswag). This specialization underscores the value of targeted instruction tuning for domain-specific excellence. Critically, human evaluators, native speakers attuned to local nuances, confirmed that our models capture cultural subtleties, idiomatic expressions, and contextual relevancies that generic multilingual models often miss (3.51 vs 2.61). This human validation provides the most compelling evidence of practical utility. The results demonstrate that targeted Urdu instruction tuning transforms commodity multilingual models into culturally competent specialists, achieving state-of-the-art performance with computational efficiency suitable for low-resource deployment.

Limitations

Our model was trained on a limited portion of the Urdu dataset due to computational and cost constraints. As a result, it exhibits gaps in knowledge as evident in evaluation on benchmark datasets.

While this version serves as a foundational step, its full potential can only be unlocked with access to a more extensive dataset to enhance its contextual understanding.

Additionally, detoxification processes were not incorporated during training, leaving the model uncensored and potentially prone to generating harmful or offensive content, which requires caution during deployment.

Evaluating LLMs also presents a significant challenge, especially for underrepresented languages like Urdu, due to the lack of standardized benchmarks outside the European linguistic domain. Although this paper introduces a tailored evaluation approach for Urdu LLM evaluation for local content, it relies on translated benchmarks to assess the model's performance across diverse applications. Translation errors may therefore have affected the results.

Ethics Statement

This research utilizes publicly available, open-source datasets that do not contain personal or identifiable information, ensuring no associated risks. All work and ideas presented are original, with AI models used solely for grammatical correction and writing enhancement. Proper citations have been made for all models and datasets used. Moreover as a generative model, it retains the potential to generate harmful or offensive content if prompted inappropriately, underscoring the need for responsible usage and careful oversight during deployment.

8. References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4344–4355, Marseille, France. European Language Resources Association.

Mohammad Amin Abbasi, Arash Ghafouri, Mahdi Firouzmandi, Hassan Naderi, and Behrouz Minaei Bidgoli. 2023. Persianllama: Towards building first persian large language model.

Anthropic. 2023. Model card and evaluations for claude models.

Samee Arif, Abdul Hameed Azeemi, Agha Ali Raza, and Awais Athar. 2024. Generalists vs. specialists: Evaluating large language models for Urdu. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7263–7280, Miami, Florida, USA. Association for Computational Linguistics.

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

Hao Chen, Abdul Waheed, Xiang Li, Yidong Wang, Jindong Wang, Bhiksha Raj, and Marah I. Abdin. 2024a. On the diversity of synthetic data and its impact on training large language models.

Jie Chen, Zhipeng Chen, Jiapeng Wang, Kun Zhou, Yutao Zhu, Jinhao Jiang, Yingqian Min, Wayne Xin Zhao, Zhicheng Dou, Jiaxin Mao, Yankai Lin, Ruihua Song, Jun Xu, Xu Chen, Rui Yan, Zhewei Wei, Di Hu, Wenbing Huang, and Ji-Rong Wen. 2024b. Towards effective and efficient continual pre-training of large language models.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhanta Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, 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. *arXiv preprint arXiv:2210.11416*.

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. *arXiv preprint arXiv:1803.05457*.

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.

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

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning.

Philipp Ennen, Po-Chun Hsu, Chan-Jan Hsu, Chang-Le Liu, Yen-Chen Wu, Yin-Hsiang Liao, Chin-Tung Lin, Da-Shan Shiu, and Wei-Yun Ma. 2023. Extending the pre-training of BLOOM for improved support of traditional chinese: Models, methods and results.

Jay Gala, Thanmay Jayakumar, Jaavid Aktar Husain, 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.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the Leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 759–765, Istanbul, Turkey. European Language Resources Association (ELRA).

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 Cucurull, 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, Zhang Guangyi, 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, Wang Yu, 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.

Tahmid Hasan, Abhik Bhattacharjee, Md Saiful Islam, Kazi Samin, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages.

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

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

Shengchao Hu, Li Shen, Ya Zhang, Yixin Chen, and Dacheng Tao. 2023. On transforming reinforcement learning by transformer: The development trajectory.

H. Huang, F. Yu, J. Zhu, X. Sun, H. Cheng, D. Song, Z. Chen, A. Alharthi, B. An, J. He, Z. Liu, Z. Zhang, J. Chen, J. Li, B. Wang, et al. 2023. Acegpt: Localizing large language models in arabic. *arXiv preprint arXiv:2309.12053*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chap-

lot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. 2022. Deduplicating training data mitigates privacy risks in language models.

Mahwiz Khalil. 2024. Urdu alpaca filtered dataset. Accessed: 2025-02-07.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina McMillan-Major, Gerard Dupont, Stella Biderman, Anna Rogers, Loubna Ben allal, Francesco De Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Mu noz, Jian Zhu, Daniel Van Strien, Zaid Alyafeai, Khalid Almubarak, Minh Chien Vu, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Adelani, Long Phan, Hieu Tran, Ian Yu, Suhas Pai, Jenny Chim, Violette Lepercq, Suzana Ilic, Margaret Mitchell, Sasha Alexandra Luccioni, and Yacine Jernite. 2022. The bigscience roots corpus: A 1.6tb composite multilingual dataset. In *Advances in Neural Information Processing Systems*, volume 35, pages 31809–31826. Curran Associates, Inc.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the Machine Learning Research (PMLR)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

W. Lian, B. Goodson, E. Pentland, A. Cook, C. Vong, and Teknium. 2023. Openorca: An open dataset of gpt-augmented flan reasoning traces.

Yen-Ting Lin and Yun-Nung Chen. 2023. Taiwan llm: Bridging the linguistic divide with a culturally aligned language model.

Yuting Lu, Chao Sun, Yuchao Yan, Hegong Zhu, Dongdong Song, Qing Peng, Li Yu, Xiaozheng Wang, Jian Jiang, and Xiaolong Ye. 2024. A comprehensive survey of datasets for large language model evaluation. In *2024 5th Information Communication Technologies Conference (ICTC)*, pages 330–336.

Shahzad Nazir, Muhammad Asif, Mariam Rehman, and Shahbaz Ahmad. 2024. Machine learning based framework for fine-grained word segmentation and enhanced text normalization for low resourced language. *PeerJ Computer Science*, 10:e1704.

Quan Nguyen, Huy Pham, and Dung Dao. 2023. Vinallama: Llama-based vietnamese foundation model.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024a. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.

Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Zhiqiang Hu, Chenhui Shen, Yew Ken Chia, Xingxuan Li, Jianyu Wang, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, Chaoqun Liu, Hang Zhang, and Lidong Bing. 2024b. SeaLLMs - large language models for Southeast Asia. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 294–304, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2024. Gpt-4o mini: Advancing cost-efficient intelligence.

OpenAI. 2025. Gpt-5 nano model. https://platform.openai.com/docs/models/gpt-5-nano.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only.

PyTorch Contributors. 2024. torchtune: Pytorch-native library for llm fine-tuning.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *Proceedings of the IEEE Conference on High Performance Computing, Networking, Storage, and Analysis*.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506, New York, NY, USA. Association for Computing Machinery.

Aaqib Saeed. 2023. Databricks dolly 15k urdu dataset. Accessed: 2025-02-07.

SarvamAI. 2023. Openhathi series: An approach to build bilingual llms frugally.

Hafiz Muhammad Shafiq, Bilal Tahir, and Muhammad Amir Mehmood. 2020. Towards building a urdu language corpus using common crawl. *Journal of Intelligent & Fuzzy Systems*, 39(2):2445–2455.

Muhammad Ali Shafique, Kanwal Mehreen, Muhammad Arham, Maaz Amjad, Sabur Butt, and Hamza Farooq. 2025. Alif: Advancing urdu large language models via multilingual synthetic data distillation. *arXiv preprint arXiv:2510.09051*.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Swabha Swayamdipta, Hyung Won Lee, Faisal Ladhak, Hexiang Tan, Prakhar Mishra, Yulia Tsvetkov, Mike Lewis, and Yequan Gu. 2023. Language models are multilingual chain-of-thought reasoners. *International Conference on Learning Representations*. Accepted at ICLR 2023.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

Munief Hassan Tahir, Sana Shams, Layba Fiaz, Farah Adeeba, and Sarmad Hussain. 2025. Benchmarking the performance of pre-trained llms across urdu nlp tasks. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 17–34, Lahore, Pakistan. International Committee on Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yannick Dubois, Xi Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

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. *arXiv preprint arXiv:2212.10560*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Sha Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, Xiao Liu, Yukuo Cen, Xu Zou, Zhilin Yang, and Jie Tang. 2021. Wudaocorpora: A super large-scale chinese corpora for pre-training language models. *AI Open*, 2:65–68.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can

a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Wei Zeng, Xiaozhe Ren, Teng Su, Hui Wang, Yi Liao, Zhiwei Wang, Xin Jiang, ZhenZhang Yang, Kaisheng Wang, Xiaoda Zhang, Chen Li, Ziyan Gong, Yifan Yao, Xinjing Huang, Jun Wang, Jianfeng Yu, Qi Guo, Yue Yu, Yan Zhang, Jin Wang, Hengtao Tao, Dasen Yan, Zexuan Yi, Fang Peng, Fangqing Jiang, Han Zhang, Lingfeng Deng, Yehong Zhang, Zhe Lin, Chao Zhang, Shaojie Zhang, Mingyue Guo, Shanzhi Gu, Gaojun Fan, Yaowei Wang, Xuefeng Jin, Qun Liu, and Yonghong Tian. 2021. Pangu-alpha: Large-scale autoregressive pretrained chinese language models with auto-parallel computation.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model.

Zhe Zhao, Yudong Li, Cheng Hou, Jing Zhao, Rong Tian, Weijie Liu, Yiren Chen, Ningyuan Sun, Haoyan Liu, Weiquan Mao, Han Guo, Weigang Gou, Taiqiang Wu, Tao Zhu, Wenhang Shi, Chen Chen, Shan Huang, Sihong Chen, Liqun Liu, Feifei Li, Xiaoshuai Chen, Xingwu Sun, Zhanhui Kang, Xiaoyong Du, Linlin Shen, and Kimmo Yan. 2023. TencentPretrain: A scalable and flexible toolkit for pre-training models of different modalities. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 217–225, Toronto, Canada. Association for Computational Linguistics.