Does LLM Quantization solve the Low Resource Double Bind for Urdu?

Anonymous ACL submission

Abstract

Many communities across the globe face a *"low-resource double bind"*: limited computing power and scarce local-language data for local LLM development. Model compression techniques such as quantization are proposed as a solution, but the performance of quantized LLMs on low resource languages - especially non-Latin scripts like Urdu - remains underexplored. In this study we evaluate the performance of eight quantized small LLMs - including Gemma3, LLama3.1 and GPT-oss - on eight specific Urdu generation and classification tasks. While GPT-oss leads in classification and Llama3.1 dominates in generation, we find all models exhibiting highly volatile performance and falling short of benchmarks across tasks. We further trace these results to the models' training datasets and architectures, and find that smaller models like Gemma3 can perform at par with models 10 times their size when pretrained on multilingual corpora. Finally, we present a quality evaluation of Urdu benchmarking datasets and scores, highlighting critical flaws and suggesting design principles for more faithful evaluations. Our study demonstrates that LLM quantizations can risk creating second-tier AI for low-resource communities in the absence of authentic evaluations on individual low-resource languages.

1 Introduction

Quantization is a model compression technique that is increasingly being explored as a pathway for edge AI in low-resource environments, allowing Large Language Models (LLMs) to be deployed on devices with limited compute and storage capacity. Model quantization converts high-precision floating-point numbers (e.g. 32-bit floats) that represent model weights and activations into low-precision integers (e.g, 8-bit integers) to achieve smaller model size and faster inference speeds with minimal accuracy loss. The technique can be applied in one of two settings - during training i.e. Quantization Aware Training (QAT) or after training i.e. Post Training Quantization (PTQ). PTQ is particularly interesting because it offers the advantage of compressing large multilingual models that perform well on low-resource languages and package them into offline low-compute versions, which is especially relevant to the context of communities that face the *low-resource double bind* (Ahia et al., 2021)

The low-resource double bind is the phenomenon wherein low-resource countries such as those in the Global South are limited in their ability to create their own localized LLMs as they possess neither the compute nor the high quality local language datasets to train LLMs. Quantization thereby emerges as a key solution to this challenge of socially equitable LLM computing by enabling edge-AI. In recent times, thousands of quantized models have been released on HuggingFace (Hugging Face, 2025) and deployed in practical use cases, including the biomedical industry (Zhan et al., 2025). While numerous evaluations of state-of-the-art frontier LLMs have been conducted (Jin et al., 2024b) (Arif et al., 2024a) (Sindhujan et al., 2025), there is surprisingly limited literature measuring the performance of quantized LLMs, especially on low-resource languages. In fact, a study reports that the performance of quantized models degrades the most for non-Latin scripts (Marchisio et al., 2024), which overlap with many low-resource languages such as Urdu. Without re-centering evaluations of quantized LLMs on these communities, this current landscape risks creating "second-tier AI" for low-resource communities, exaggerating their double-bind.

With this objective in this study, we evaluate the performance of eight diverse mainstream quantized LLMs on Urdu - a low-resource language native to Pakistan, with 230 million speakers worldwide. Pakistan has over 120 million Urdu speakers while ranking 67th out of 116 on the English Proficiency

index (EF) (EF Education First, 2024) , resulting in Urdu being the most accessible lingua franca for the region. However, Urdu remains low-resource because of a critical lack of digital assets, including large-scale annotated corpora.

We evaluate models across eight generation and classification tasks, to report the true accessibility of quantized models for Urdu populations. Evaluating these models, we uncover another dimension of the double bind – poor quality *benchmark* datasets and inadequate scoring schemes, apart from training datasets. Hence, we present a detailed report on the flaws in state-of-the-art Urdu evaluation schemes, and suggest schemes that are more faithful.

Our main contributions in this study are as follows:

1. **Benchmarking:** A comprehensive evaluation of quantized language models on Urdu tasks.

2. **Analysis:** Analysis of performance differences in terms of model architectures and training datasets, identifying configurations that enhance Urdu understanding.

3. **Quality Assessment:** Quality evaluation of existing Urdu benchmarking datasets and evaluation schemes.

4. **Recommendations:** Best practices for the design and evaluation of quantized models for low-resource languages, particularly Urdu.

2 Related Work

2.1 Low Resource Languages

Large Language Models (LLMs) have been widely reported to perform poorly on Low Resource Languages (LRLs), owing to a multitude of shortcomings in the LLM pipeline - from training datasets to tokenizers to fine-tuning mechanisms. LRLs, especially those like Urdu, are morphologically rich with complex grammar such as agglutinative properties, which English-trained LLMs fail to generalise to. A study investigating translation capabilities of GPT models showed that the models achieve competitive scores when translating high-resource languages like English, but only show limited performance on low-resource languages such as Hausa and Icelandic without any significant improvement upon few-shot prompting (Hendy et al., 2023). Another study identified and categorized "curses" of LLMs for low-resource languages into two categories: "harmfulness" and "relevance", quoting that GPT-4 produced harmful responses upon malicious prompting 35% of the time for LRLs compared to 1% for HRLs and irrelevant content in 20% responses for LRLs versus 0% in HRLs (Shen et al., 2024).

Studies investigating Urdu in particular also revealed that mainstream models including GPT and Llama failed to meet the SOTA benchmark on 18 out of 20 Urdu tasks such as emotion classification and fake news detection (Tahir et al., 2025) whereas another study demonstrated GPT-4 achieving around 13% lesser F_1 scores for Urdu on question-answering as compared to English. (Kazi et al., 2025). Another study evaluating LLM-based translation quality estimation for low-resource languages including Hindi (mutually intelligible with Urdu) and Arabic (similar script as Urdu) showed that Latin-script languages such as English achieved better performance due to efficient tokenization as compared to non-Latin scripts like Arabic, while the task was overall particularly difficult for morphologically complex languages that lack sufficient pre-training data (Sindhujan et al., 2025). Finally, it was also shown that large "generalist" models like GPT-5 perform poorly compared to smaller, LRL fine-tuned "specialist" models, because of sparse representation of LRLs in their training datasets - establishing that frontier LLMs underperform on low resource languages without necessary downstream processing (Arif et al., 2024a). This has necessitated development of fine-tuned Urdu LLMs such as UrduLLaMA (Fiaz et al., 2025), Lughaat (Noman, 2025) and Alif (large traversaal, 2025) which show significant performance improvements over their base models such as Llama.

2.2 Quantized LLMs Evaluation

Model quantization schemes are broadly categorized into Post-Training Quantization (PTQ) and Quantization Aware Training (QAT). While QAT trains with the objective of optimal accuracy upon quantization, PTQ methods suffer from performance degradation termed as the quantization loss. A study found aggressive 2-3 bit quantization producing up to 32.39% accuracy degradation on mathematical reasoning tasks in Llama-3 models (Li et al., 2025) while another study found LLMs losing instruction-following abilities when quantized to 2 bits (Jin et al., 2024a). A multi-dimensional evaluation of two quantized LLMs across 23+ languages

revealed differential degradation across tasks compared to the original model, with mathematical tasks degrading up till 17.3% and multilingual comprehension using the Belebele benchmark showing a 8.5% degradation (Marchisio et al., 2024). The evaluation also showed a disproportionate degradation across languages, particularly showing that the degradation for non-Latin scripts was 3.7% against 3.0% for Latin scripts. This is an important finding as non-Latin scripts also coincide with some low-resource languages such as Arabic and Urdu.

While another section of literature evaluates quantized LLMs on other benchmarks including efficiency, alignment and memory (Zhao et al., 2025), there is limited literature on thorough knowledge and capacity evaluations of quantized LLMs on individual low resource languages across tasks, with a critical gap for Urdu. Existing studies often evaluate quantizations of larger models (>8B) and/or a small number of models, while also sometimes fine-tuning the models before quantization. To our knowledge, ours is the first study to evaluate a diverse range of mainstream LLMs' quantizations across multiple tasks specifically for Urdu.

3 Methodology

3.1 Datasets

We use the datasets directly from the repository created by Arif et al. (2024a). Details of each dataset are as follows.

Generation

- **Question-Answering**. A collection of three datasets: (1) UQA (Arif et al., 2024b), an Urdu translation of the Stanford Question Answering Dataset (Rajpurkar et al., 2016), (2) UQuAD[1], built using various sources including Wikipedia articles and YouTube videos, and (3) Wiki-UQA[2], generated entirely from Wikipedia articles. UQuAD consists of only answerable questions, while UQA and Wiki-UQA also contain unanswerable questions, with all three of them based on extractive question-answering.

- **Transliteration**. The Dakhsina dataset by Google (Roark et al., 2020), consisting of native script Wikipedia text along with its romanization.

- **Translation**. OPUS-100 (Zhang et al., 2020; Tiedemann, 2012), a sampling of the OPUS corpus containing English-Urdu sentence pairs sourced from multiple corpora including movie subtitles and GNOME documentation.

- **Summarization**. XSUMUrdu (Munaf et al., 2024), containing abstractive and extractive summaries of data scraped from the BBC Urdu and DW Urdu News websites.

Classification

- **Sentiment Analysis**. The Urdu IMDB dataset, created by translating IMDB rmovie reviews (Maas et al., 2011) into Urdu using Google Translator. The sentiment labels are only positive or negative.

- **Topic Classification**. The Urdu-News-Headline-Dataset[3], consisting of news headlines and summaries of digital Urdu news articles, categorised into eight topics: business, entertainment, health, politics, science, sports, world and other. For our task, we use the headlines as our input.

- **Named Entity Recognition (NER)**. The urdu-ner dataset on HuggingFace[4], containing NER-tags for each token in Urdu sentences. There are eight tags: person, organization, number, location, designation, date, time, and other. One token per entry to be classified is wrapped in `<h1>` tag for out prompt.

- **Part-of-Speech (PoS) Tagging**. The Universal Dependencies dataset (Nivre et al., 2020) for Urdu, annotated with a PoS-label for each token in sentences. Target tokens are wrapped in the `<h1>` tag similarly to NER. There are 19 labels, shown in Table 9.

3.2 Models

We evaluate a selection of recent, instruction-tuned small Large Language Models (LLMs) from diverse model families, encompassing a variety of model architectures and training datasets. These models represent the cutting edge in open-source AI, each with unique strengths and design philosophies. The selected models with their training data size and composition are listed in Table 1. For simplicity

[1] https://github.com/ahsanfarooqui/UQuAD—Urdu-Question-Answer-Dataset/tree/main

[2] https://huggingface.co/datasets/uqa/Wiki-UQA

[3] https://github.com/mwaseemrandhawa/Urdu-News-Headline-Dataset

[4] https://huggingface.co/datasets/mirfan899/urdu-ner

throughout the rest of the paper, we refer to the models without their version numbers, except for Gemma models.

3.3 Model Setup

For our model setup, we use `llama.cpp`[5] due to its widespread support for on-device inference across models as well as edge devices. `llama.cpp` uses the `.gguf` file format with multiple quantization levels such as `Q4_K_M`, `Q5_K` and `Q6_K` – each with its own tradeoffs. For our experiments, we use the `Q4_K_M` quantizations as they offer the best balance between speed, size and performance (perplexity and accuracy). The `Q4_K_M` level uses 4-bit, group-wise quantization with medium-precision. We directly downloaded the `Q4_K_M.gguf` file for each of our models from HuggingFace and did not perform any quantization ourselves. The files are listed in Appendix E. We utilised GPU offloading support of `llama.cpp` to run our evaluations faster while preserving the output quality.

We host our models on llama-server and use the `/v1/chat/completions` API to fetch responses. We set the temperature to `0.0` and let `llama.cpp` use default values for other sampling parameters such as top-k and top-p.

3.4 Experimental Design

We benchmark the performance of the models on each task using standard metrics. For all the classification tasks, we compute the Macro-F_1. We use SQuAD F_1 (Rajpurkar et al., 2016) (Rajpurkar et al., 2018) for question-answering, ROUGE-L (Lin, 2004) with word-level tokenization for summarization, SacreBLEU (Post, 2018) for transliteration and translation. Additionally for generation tasks, we exclude responses where the model produced output containing more than 50% English characters, representing the model answering almost entirely in English instead of Urdu. We assume non-English characters to be Urdu, based on the observation that model outputs were ever entirely in another language except English, although they often produced a few Chinese, Hindi or Russian tokens in the middle of sentences. We do not filter based on the percentage of Urdu characters however, to accommodate the various Unicode representations and diacritic variants commonly present in Urdu text. While we evaluate all tasks using the original dataset and labels, we evaluate two tasks - question-answering and PoS-tagging - in two different settings.

- **Question-Answering: Abstain vs No-Abstain.** For the first setting in question-answering, we prompt the model to abstain from answering if the answer is not found in the text, while in the other we remove this explicit instruction.

- **PoS-tagging: Merged vs Unmerged Labels.** For PoS tagging, we create a variant of the original dataset where we merge the labels into broader categories. This is based on the hypothesis that smaller models may struggle to differentiate micro-categories within broader ones. The merged categories were constructed as follows:

  1. subordinating conjunction, coordinating conjunction → conjunction
  2. symbol, _, punctuation mark → symbol
  3. proper noun, noun → noun
  4. auxiliary verb, verb → verb

3.5 Prompt Design

As reported in Khondaker et al. (2023) and referenced in Arif et al. (2024a), English prompts were observed to perform better than the Arabic ones while benchmarking ChatGPT on Arabic - a low-resource language. Therefore, we also design all our prompts in English for evaluating Urdu performance. To create prompts for evaluating small quantized LLMs on low-resource languages, we define the following design principles:

- *Concise*. We keep the prompt length minimal, accounting for the relatively smaller context windows of small LLMs.

- *Precise*. Instead of using generic prompts, we design context-specific ones tailored to the dataset. For example, instead of *"classify the given Urdu text into one of the following **labels**…"*, we use *"classify the given Urdu news text into one of the following **topics**…"*. This is aimed at reducing model confusion as much as possible while interpreting a low-resource language and improving reasoning accuracy.

- *Simple*. We keep prompts grammatically simple. For example, instead of *"classify the given movie review into one of the following sentiment labels: ['positive', 'negative']"*, we use

[5] https://github.com/ggml-org/llama.cpp

Model	Size (billion parameters)	Training Dataset
Qwen2.5	1.78	Primarily English (Yang et al., 2024)
Gemma 2	2.61	Primarily English (Team, 2024)
Gemma 3	3.88	Multilingual (Team, 2025)
Phi3.5-mini	3.82	Multilingual (Abdin et al., 2024)
Mistral-7B	7.42	Primarily English (Jiang et al., 2023)
Deepseek-R1-Distill-Qwen	7.62	Primarily English (Guo et al., 2025)
Llama 3.1	8.03	Multilingual with Urdu (Grattafiori et al., 2024)
GPT-oss	20.9	Multilingual (OpenAI, 2025)

Table 1: Model Quantizations Evaluated

"classify the given movie review as 'positive' or 'negative' based on its sentiment". This attempts to lighten the model's limited "cognitive load" or attention from unnatural syntax parsing and focusing it towards instruction understanding.

Following these principles, we design zero-shot prompts structured into a given template shown in the Appendix C.

4 Results

We analyze the performance across tasks of each model against its size, and put these results into context using a baseline. The baseline value selected was the performance of GPT-4 Turbo with zero-shot prompting on the same datasets, as reported in Arif et al. (2024a). We consider it a lenient benchmark because it represents the state-of-the-art from the best general model in 2024. The results are shown in Table 2 and Figure 1.

4.1 Generation

For all generation tasks except summarization, Llama outperformed all models scoring 57.96 SQuAD-F_1 on Question-Answering, 21.95 SacreBLEU on transliteration, 14.67 SacreBLEU on ur-en Translation, and 6.59 SacreBLEU on en-ur Translation. For summarization, Gemma2 achieved the highest Rouge-L score of 0.24. We observe the performance of Gemma3 to be at par with Llama despite being half the model size, especially on tasks such as Transliteration and ur-en Translation where Gemma3 scores 20.73 and 6.30 SacreBLEU respectively. Gemma2 also joins this high-performing pair on Question Answering and ur-en Translation, with the performance difference between Gemma2 and Gemma3 being 0.44 and 1.50 respectively. For other tasks, Gemma2 lies closer to a mid to low-performing cluster comprised of Phi, Mistral, Deepseek, Qwen, and GPT. While Qwen, Phi and Mistral achieve mid-tier performance on tasks such as Question-Answering and Transliteration, the others in the low-performing cluster perform below 50% of the performance of Llama on all tasks except summarization. The performances of these models fall as low as SacreBLEU 0.14 for Deepseek and 0.611 for GPT on Transliteration, as well as 0.0392 for Qwen on ur-en Translation.

Performance close to the benchmark is achieved on all tasks except Translation (en-ur), with Δ=8.32 for Question-Answering, Δ=8.38 for Transliteration, Δ=1.62. For Translation (en-ur), the maximum performance of Llama remains at 58% of the benchmark. In summarization, the benchmark is surpassed by Gemma2 achieving a performance of 0.24 Rouge-L against the 0.22 benchmark. Our findings on translation are also consistent with prior work (Sindhujan et al., 2025, Nguyen et al., 2024) showing that scores are higher when English is the target language, and the low-resource double bind is prominent in the opposite scenario.

No clear correlation was observed between model size and performance for generation - the largest model i.e. GPT performed poorly while any given size category also featured significant performance differences, with Llama and Deepseek as examples.

4.2 Classification

Across all classification tasks, GPT consistently outperforms other models, achieving the highest Macro-F_1 scores on all tasks - 0.63 on Topic Classification, 0.87 on Sentiment Analysis, 0.54 on NER, and 0.42 on POS Tagging. Its performance closely matches the benchmark, with deltas of 0.13, 0.03, 0.07, and 0.11 respectively, demonstrating robust generalization across multiple classification tasks of different label set sizes. GPT is closed followed by

Task	Qwen2.5	Gemma 2	Phi3.5-mini	Gemma 3	Mistral-7B	Deepseek	Llama 3.1	GPT-oss
QA (abstain)	27.60	50.91	27.54	43.23	13.90	4.80	**71.39**	15.76
QA (no-abstain)	38.07	47.16	36.12	47.60	26.66	16.86	**57.96**	23.12
Transliteration	1.35	9.85	2.66	20.73	5.13	0.14	**21.95**	0.61
Translation (ur-en)	3.05	10.12	2.54	11.62	6.55	3.02	**14.67**	3.70
Translation (en-ur)	0.04	3.21	0.40	6.30	0.62	0.24	**6.59**	3.00
Summarization	0.21	**0.24**	0.19	0.21	0.16	0.15	0.08	0.20
Topic Classification	0.35	0.53	0.28	0.57	0.40	0.48	0.40	**0.63**
Sentiment Analysis	0.60	0.79	0.31	0.84	0.42	0.74	0.66	**0.87**
NER	0.35	0.35	0.35	0.43	0.35	0.40	0.11	**0.54**
POS-Tagging (Merged)	0.18	0.13	0.21	0.24	0.35	0.18	0.24	**0.42**
POS-Tagging (Unmerged)	0.15	0.05	0.18	0.17	0.21	0.12	0.16	**0.42**

Table 2: Evaluation results. Eight quantized models are evaluated across eight generation and classification tasks on Urdu.

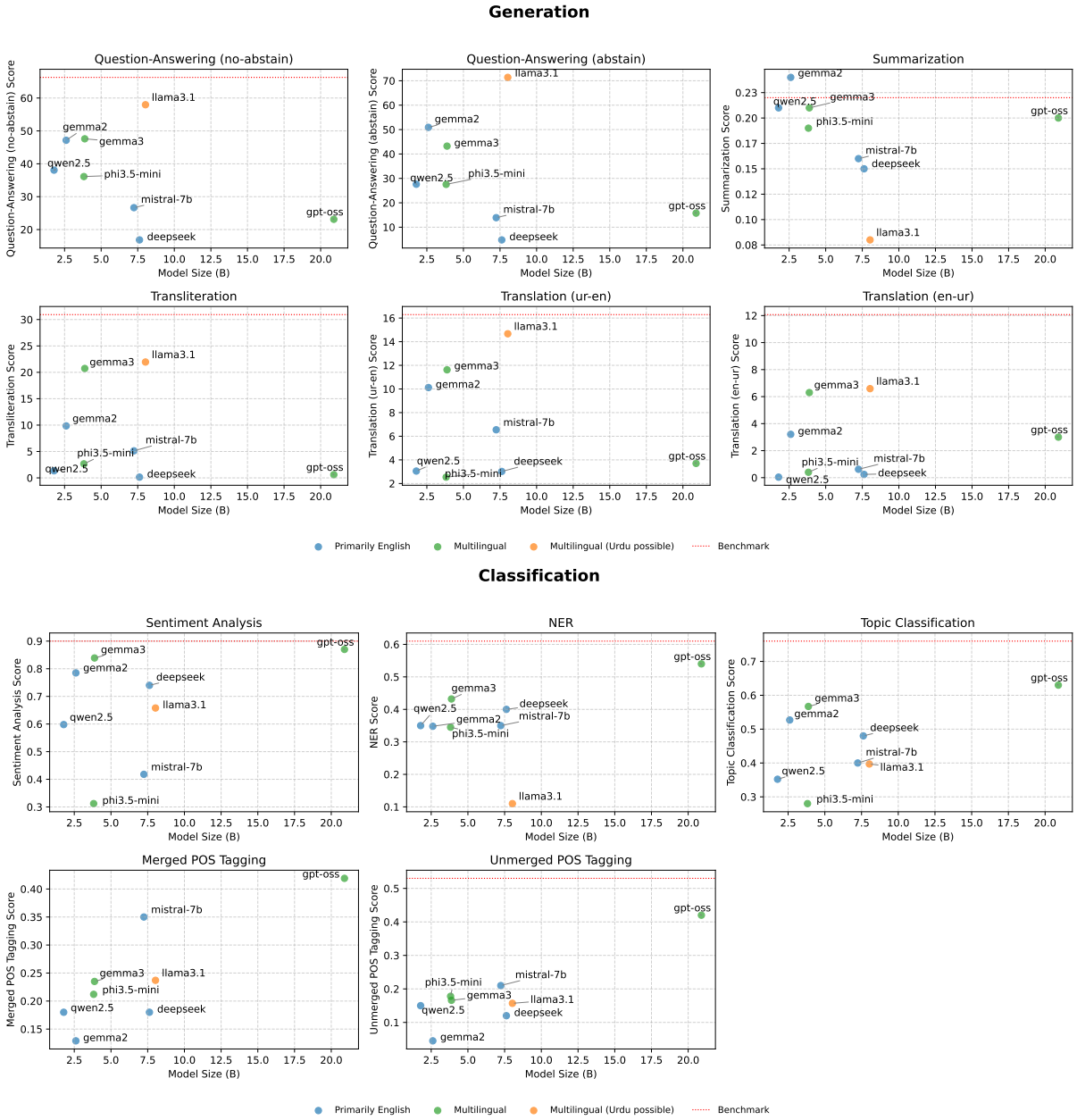

Figure 1: Model performance against model size in billion parameters on Urdu Generation and Classification tasks.

Gemma2 and Gemma3, particularly in Topic Classification and Sentiment Analysis, where Gemma3 shows a difference of only 0.03 and 0.06 points, despite being less than one-fifth the size of GPT. A

mid-tier cluster is observed comprised of Deepseek, Qwen, and Mistral, all scoring between 0.35–0.48 on topic classification and 0.41–0.74 on sentiment. Llama and Phi form a low performing tier with Macro-F_1 scores as low as 0.11 for Llama on NER and 0.28 for Phi on Topic Classification.

A contrast is observed between token-level tasks (NER and POS-tagging) and text-level understanding tasks (Topic and Sentiment Classification). Token-level tasks feature all models clustering in similar performance ranges, suggesting a shared difficulty in token-level classification, whereas higher variability is observed in scores across models for text-level tasks, suggesting higher model sensitivity to architecture and training datasets for text understanding.

While model size generally correlates with performance, there are notable exceptions such as Gemma2 surpassing models 10 times its size - Mistral and Deepseek - on Topic Classification and Sentiment analysis.

All models achieve over 50% of benchmark performance on Topic Classification, Sentiment Analysis, and NER; however, for POS tagging, all except GPT fall below the 50% threshold, a challenge we attribute to the micro-labels in the label list and explored in the special setups section.

4.3 Model Variance

Examining the ranks of each model across each task as shown in Figure 4 reveals that the models show highly inconsistent performance. While Gemma3 is the only model performing at a constant high ranking across most tasks, other models such as Phi and Mistral show high variance between tasks, fluctuating between high and low ranks. Figure 2 further clarifies this high performance irregularity, showing that models that offer good performance on a certain type of tasks (generation/classification) often fail drastically at the converse set of tasks - implying a classification-generation performance tradeoff.

Analysing the average generation and classification scores of models as shown in Figure 3, we observe all models performing better on classification with drastic performance differences with generation. While Llama remains the only model with a balanced performance on both tasks, all other models demonstrate a 1.5x or higher performance on classification, up to as high as 3x for Mistral, 5.75x for Deepseek, and 5.2x for GPT.

A notable phenomenon is observed for Llama and GPT - the top performers. These models maintain the highest rankings on one type of task (generation or classification) but drop to the lowest ranks for the other task. Inspection of model predictions revealed that both models were facing degenerative looping - with Llama repeatedly outputting all labels instead of one for POS-tagging, and GPT repeating the same thinking until the output token limits ran out for generation tasks.

This classification-generation tradeoff can be explained in terms of the testing objectives of each type of task and the nature of model embeddings. Generation tasks probe token-level comprehension of Urdu morphology and semantics, while classification tasks require discourse-level understanding of the entire text, indicating that models perform better at text-level than the token-level. This can be attributed to our classification labels being in English, enabling cross-lingual transfer via the label embeddings trained across large web corpora in multiple languages. On the contrary, generation tasks require Urdu specific training, which was missing in the training corpora of the models. Within classification, NER and POS also analyse on the token-level. While POS did emerge as the most challenging task, NER saw high performance across all models - this can be attributed to POS tags being largely independent of the whole text and focused on the single token only whereas NER is highly context dependent to classify the single token. Hence, these results only further solidify our understanding of model abilities being superior on whole text over tokens in Urdu.

4.4 Special Setups

Abstain vs no Abstain. Models with strong performance on the Question Answering task - mainly LLaMA 3.1 and Gemma 2 - achieve higher accuracy when asked to abstain if the answer is not present in the passage. By contrast, models with weaker performance - such as Mistral, DeepSeek, and Phi - exhibit reduced accuracy under the abstention condition. These findings suggest that high-performing models display better calibration, demonstrating greater confidence in their correct answers relative to their incorrect ones which are more likely to be abstained from to improve overall performance. Conversely, low-performing models tend to lack confidence even in their correct or partially correct responses, leading them to abstain unnecessarily and thus reducing overall accuracy. The results are shown in Figure 5 in Appendix A.

Merged vs Unmerged POS-tagging. Merging the POS labels into broader categories resulted in improved performance across all models except GPT, which exhibited a marginal decrease of 0.001 points. The least-scoring model under the unmerged label setting - Gemma 2 - showed a 2.8-fold improvement in performance after merging. On average across all models, the ratio of merged to unmerged performance was 1.5, indicating that smaller models may struggle to reliably distinguish fine-grained POS categories in Urdu. A major factor contributing to the observed improvement is that models produced the placeholder symbol ('_') tag with notable frequency prior to label merging. The results are shown in Figure 6 in Appendix A.

5 Factors Affecting Performance

5.1 Training Dataset Size and Composition

In addition to model size, the number of training tokens and and multinguality of the training dataset are important factors affecting performance, as shown in Figure 7.

For training tokens alone, we observe no clear correlation with the task performance. The performance of Llama with 15 trillion tokens is at par with that of Gemma3 in transliteration at 4 trillion tokens. Similarly, models on the lower end of training dataset size such as Gemma2, Phi and Deepseek often outperform models on the upper end such as Llama and Qwen. Accounting for the dataset composition explains these results - among the upper end models, Llama is the only model trained on multilingual corpora, confirming that complementing a high token count (15M+) with multingual datases is crucial for good performance. The performance of Gemma3 can also be attributed to its balanced pretraining mixture (Team, 2025), explicitly optimized for multilingual robustness and low-resource language coverage.

It is also observed that Generation tasks are more sensitive to multilingual pretraining datasets. Primarily English models such Gemma2 and Mistral can perform moderately well on cross-lingual semantic transfer tasks such as Topic and Sentiment Classification but perform poorly on token-level Urdu tasks such as translation and transliteration. This confirms limited script generalization and different scaling laws for Urdu and English upon quantization. In classification tasks, the gap between the performance of multilingual and non-multilingual models is smaller, owing to the English labels leveraging cross-lingual transfer.

Overall, these results indicate that while multilingual training datasets are essential for token-level understanding such as Generation tasks, English-focused training datasets produced via general web crawls produce knowledge representations that can leverage cross-lingual transfer to generalize to low-resource languages for text-level understanding such as Classification tasks.

5.2 A Pareto Frontier

The big question our study attempts to answer is: *does making models smaller through quantization truly make them more accessible for low-resource communities?* Our data shows a general decline in performance as model size decreases, suggesting a trade-off between compute-efficiency and output quality. However, as discussed above, performance is a function of multiple variables beyond size.

We answer this question through an exciting observation: a Pareto frontier in the current state-of-the-art of quantized Urdu models (Figure 8). A Pareto frontier represents the optimal result achievable within given constraints - in our case, the best model performance possible within each size category. While Pareto frontiers are well documented for high-resource languages, our study provides the first clear characterization for quantized models on Urdu as a low-resource language.

Importantly, while the frontier establishes that there is a size–performance trade-off, our results also highlight strategies for shifting it outward. For example, we observe that the Gemma models approach the performance of the much larger Llama on generation tasks, suggesting that the performance frontier of smaller models can be advanced if they are trained on larger, more diverse datasets or fine-tuned prior to quantization.

6 Limitations

While our study focused on specific setups, its limitations highlight avenues for further exploration of quantized LLMs in low-resource contexts. First, our analysis used a single high-performance baseline (GPT-4 Turbo). A more granular approach would be to establish model-wise baselines, comparing each quantized model against its un-quantized parent. This would allow for a precise measurement of performance degradation attributable directly to the quantization process itself. Second, while model size is a conventional proxy for computa-

tional efficiency, a more user-centric metric would be inference speed (tokens per second) on common consumer hardware. Future work should measure this to provide a more practical assessment of real-world performance. Furthermore, our study was limited to a single quantization scheme (`Q4_K_M`); a comparative analysis of different quantization techniques (e.g., 2-bit, 8-bit, GGUF vs. AWQ) would reveal which methods best preserve linguistic capabilities for non-Latin scripts like Urdu. Additionally, we were unable to evaluate quantizations of Urdu fine-tuned LLMs such as UrduLLaMA (Fiaz et al., 2025), Lughaat (Noman, 2025), and Alif (large traversaal, 2025) on account of unavailability of their `.gguf` files or the performance of those available on our prompts being inadequate, requiring changes in setup that would disrupt the standardization of the experiment. Finally, and most importantly, automated metrics often fail to capture the nuances of human perception, especially for low-resource languages . A comprehensive human evaluation is essential to create a more representative assessment of model quality. Such an evaluation could also investigate critical safety issues like hallucinations and censorship. Hallucinations can misinform users, while over-aggressive censorship can erase culturally relevant discourse, both phenomena disproportionately harm vulnerable communities and warrant dedicated study.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, and et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Orevaoghene Ahia, Julia Kreutzer, and Sara Hooker. 2021. The low-resource double bind: An empirical study of pruning for low-resource machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3316–3333, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Samee Arif, Abdul Hameed Azeemi, Agha Ali Raza, and Awais Athar. 2024a. 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.

Samee Arif, Sualeha Farid, Awais Athar, and Agha Ali Raza. 2024b. Uqa: Corpus for urdu question answering. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LRECCOLING 2024)*, pages 17237–17244, Torino, Italia. ELRA and ICCL.

EF Education First. 2024. Pakistan — ef english proficiency index. https://www.ef.com/epi/regions/asia/pakistan/. Retrieved May 2025.

Layba Fiaz, Munief Hassan Tahir, Sana Shams, and Sarmad Hussain. 2025. Urdullama 1.0: Dataset curation, preprocessing, and evaluation in low-resource settings. https://arxiv.org/abs/2502.16961.

Andrew Grattafiori and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *Preprint*, arXiv:2302.09210.

Hugging Face. 2025. Hugging face: Open-source platform for machine learning. Accessed: 2025-09-12.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lelio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothee Lacroix, and William El Sayed. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Renren Jin, Jiangcun Du, Wuwei Huang, Wei Liu, Jian Luan, Bin Wang, and Deyi Xiong. 2024a. A comprehensive evaluation of quantization strategies for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12186–12215, Bangkok, Thailand. Association for Computational Linguistics.

Yiqiao Jin, Mohit Chandra, Gaurav Verma, Yibo Hu, Munmun De Choudhury, and Srijan Kumar. 2024b. Better to ask in english: Cross-lingual evaluation of large language models for healthcare queries. In *Proceedings of the ACM Web Conference 2024 (WWW '24)*, Singapore. ACM.

Samreen Kazi, Maria Rahim, and Shakeel Ahmed Khoja. 2025. Crossing language boundaries: Evaluation of large language models on Urdu-English question answering. In *Proceedings of the First Workshop on Natural Language Processing for Indo-Aryan and Dravidian Languages*, pages 141–151, Abu Dhabi. Association for Computational Linguistics.

Md Tawkat Islam Khondaker, Abdul Waheed, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Gptaraeval: A comprehensive evaluation of chatgpt on arabic nlp. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 220–247, Singapore. Association for Computational Linguistics.

large traversaal. 2025. Alif-1.0-8b-instruct: An advanced multilingual reasoning model (english & urdu). https://huggingface.co/large-traversaal/Alif-1.0-8B-Instruct.

Zhen Li, Yupeng Su, Runming Yang, Congkai Xie, Zheng Wang, Zhongwei Xie, Ngai Wong, and Hongxia Yang. 2025. Quantization meets reasoning: Exploring llm low-bit quantization degradation for mathematical reasoning. *Preprint*, arXiv:2501.03035.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Kelly Marchisio, Saurabh Dash, Hongyu Chen, Dennis Aumiller, Ahmet Ustun, Sara Hooker, and Sebastian Ruder. 2024. How does quantization affect multilingual llms? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15928–15947, Miami, Florida, USA. Association for Computational Linguistics.

Mubashir Munaf, Hammad Afzal, Khawir Mahmood, and Naima Iltaf. 2024. Low resource summarization using pre-trained language models. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 23(10):141:1–141:19.

Xuan-Phi Nguyen, Mahani Aljunied, Shafiq Joty, and Lidong Bing. 2024. Democratizing LLMs for low-resource languages by leveraging their English dominant abilities with linguistically-diverse prompts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3501–3516, Bangkok, Thailand. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Muhammad Noman. 2025. Lughaat-1.0-8b-instruct: An advanced urdu language model. https://huggingface.co/muhammadnoman76/Lughaat-1.0-8B-Instruct.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. Technical report.

Matt Post. 2018. A call for clarity in reporting bleu scores. In *Proceedings of the 3rd Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Brian Roark, Lawrence Wolf-Sonkin, Christo Kirov, Sabrina J. Mielke, Cibu Johny, Isin Demirsahin, and Keith Hall. 2020. Processing south asian languages written in the latin script: the dakshina dataset. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*, pages 2413–2423.

Lingfeng Shen, Weiting Tan, Sihao Chen, Yunmo Chen, Jingyu Zhang, Haoran Xu, Boyuan Zheng, Philipp Koehn, and Daniel Khashabi. 2024. The language barrier: Dissecting safety challenges of LLMs in multilingual contexts. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2668–2680, Bangkok, Thailand. Association for Computational Linguistics.

Archchana Sindhujan, Diptesh Kanojia, Constantin Orasan, and Shenbin Qian. 2025. When LLMs struggle: Reference-less translation evaluation for low-resource languages. In *Proceedings of the First Workshop on Language Models for Low-Resource Languages*, pages 437–459, Abu Dhabi, United Arab Emirates. 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, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Gemma Team. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Jorg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Z. Zhan, S. Zhou, M. Zeng, K. Yu, M. Song, X. Chen, J. Wang, Y. Hou, and R. Zhang. 2025. Quantized large language models in biomedical natural language processing: Evaluation and recommendation. *Preprint*, arXiv:2509.04534.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639. Association for Computational Linguistics.

Jiaqi Zhao, Ming Wang, Miao Zhang, Yuzhang Shang, Xuebo Liu, Yaowei Wang, Min Zhang, and Liqiang Nie. 2025. Benchmarking post-training quantization in llms: Comprehensive taxonomy, unified evaluation, and comparative analysis. *Preprint*, arXiv:2502.13178.

A Figures

B Benchmark Quality

While our methodology stands consistent with literature in this area, we identified notable limitations in the state-of-the-art evaluation pipeline for Urdu. In this Appendix section, we present a task-wise analysis of benchmark quality and show how they pose significant limitations to Urdu evaluation.

B.1 Translation

Dataset. The en-ur subset of the OPUS-100 dataset is comprised significantly of religious texts, with the following features:

(i) The writing is archaic and scriptural, in contrast with modern natural language. e.g. contains a lot of "thee", "thy" and Urdu equivalents.

(ii) The translations present therein contain a lot of annotations in brackets that are not translations from the text. This includes true meanings of symbolic phrases, contextual information, and vocatives. Importantly, these annotations do not always contain corresponding annotations in both languages.

(iii) Some entries (en-ur pairs) are truncated to the extent that they do not preserve grammatical or semantic sense.

(iv) Some entries are non-corresponding, i.e. the English and Urdu texts are not related to or translations of each other.

(v) A few entries do not contain Urdu, partially or completely.

Examples of these cases are shown in Table 3.

Scoring. The SacreBLEU score is calculated as an average number of exact matches across 1 to 4-grams in the predicted text against the true reference text.

Firstly, this leads to very low scores when models output more modern natural language translations that do not match the archaic translations present in the dataset, despite capturing the correct semantic meaning. This is the more frequent scenario as the model is not prompted to output in any specific style. To verify this claim, we ran a small experiment with ChatGPT as a judge, as detailed below.

- *Experiment Setup A*. We sampled 10 random en-ur pairs from the dataset and produced English translations using our most stable Urdu model Gemma 3. We then prompted ChatGPT to select the better translation between "Translation A" (dataset) and "Translation B" (predicted). We prompted with the same dataset three times.

- *Results A:* The predicted translations won in only 2/10 cases on average (2,2,2). These were instances where the reference translations were incorrect as in dataset feature (v). For the cases where reference translations won, ChatGPT stated that they were more "accurate[ly] Quranic" or "classical".

- *Experiment Setup B*. We used the same en-ur pairs and prompted ChatGPT to select the better translation, with an additional instruction: "Evaluate based on naturalness and not Quranic style". We prompted with the same dataset three times.

- *Results B:* The predicted translations won in 6/10 cases on average (5,6,8). Where predicted translations won, ChatGPT stated that they

Figure 2: Model variance across tasks. Task abbreviations: qa - question-answering, trans - translation, topic - topic classification, sentiment - sentiment analysis, pos - pos-tagging.

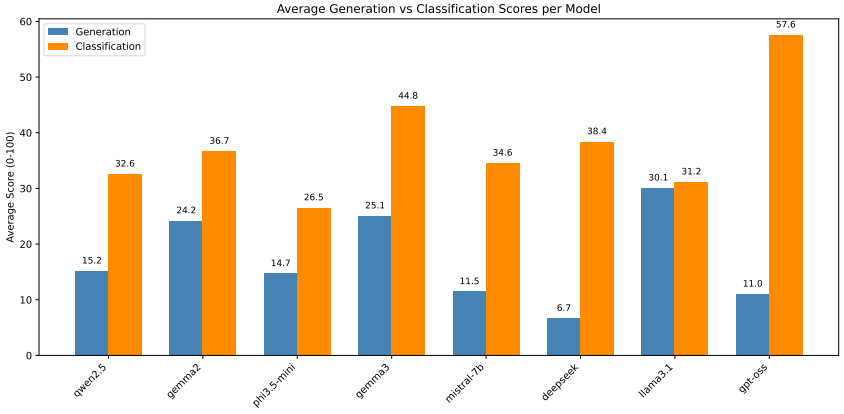

Figure 3: Average generation vs. classification scores per model.

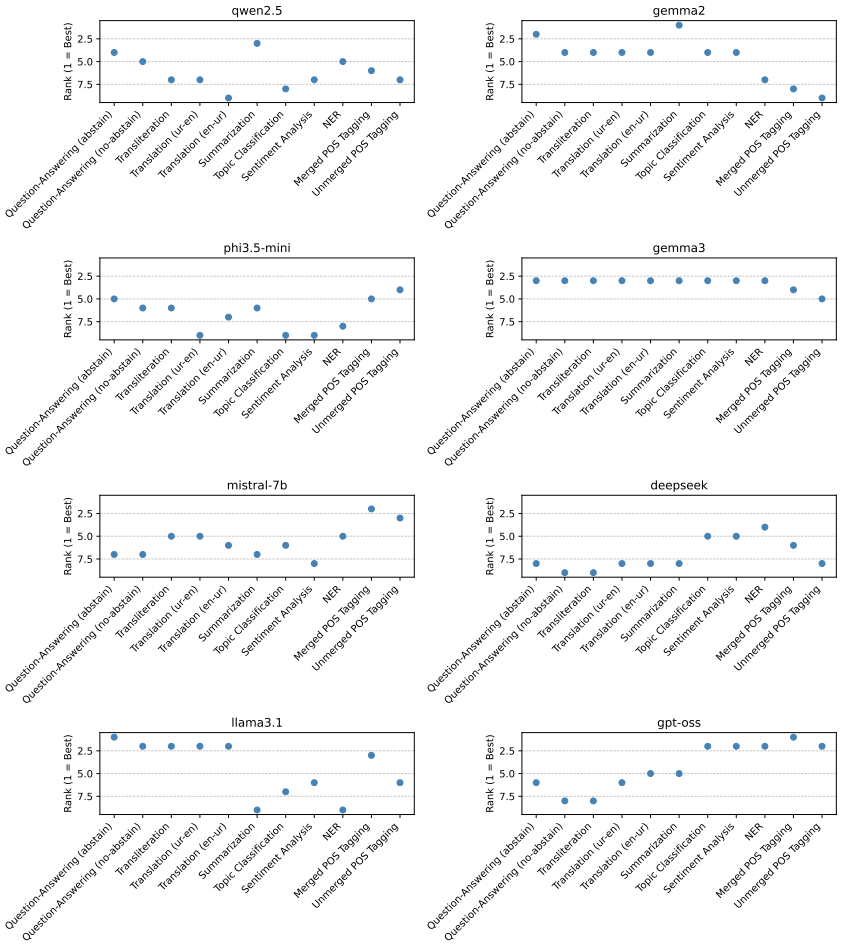

Figure 4: Model rankings across tasks.

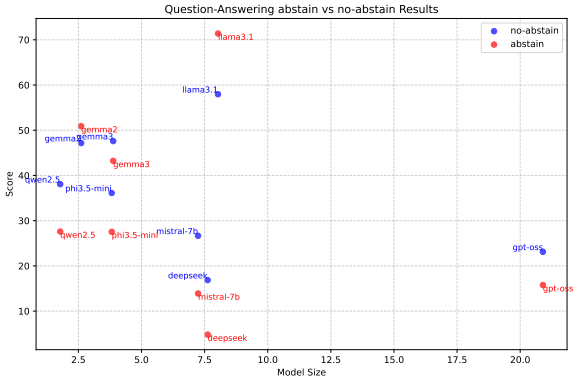

Figure 5: Question-Answering: abstain vs. no-abstain performance

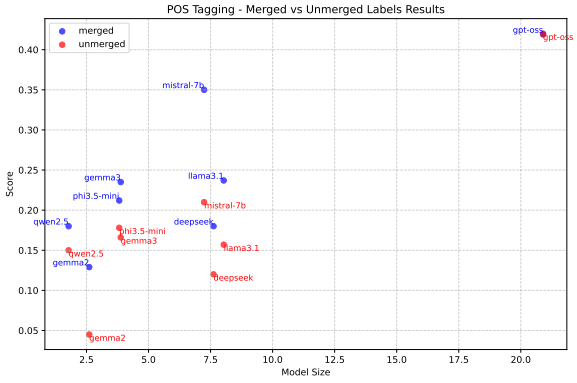

Figure 6: POS-tagging: merged vs. unmerged labels performance

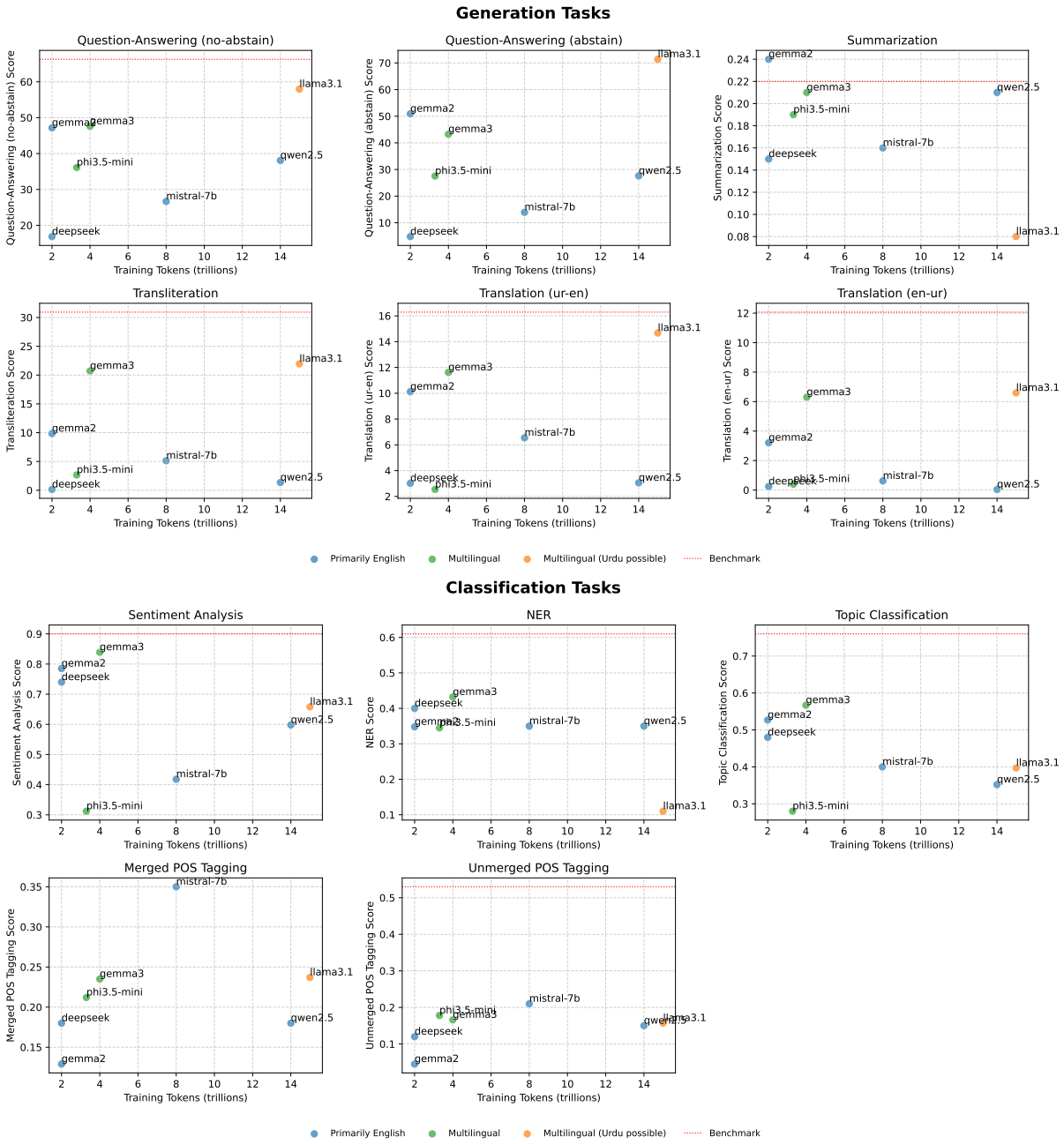

Figure 7: Model performance against training tokens on Generation and Classification tasks.

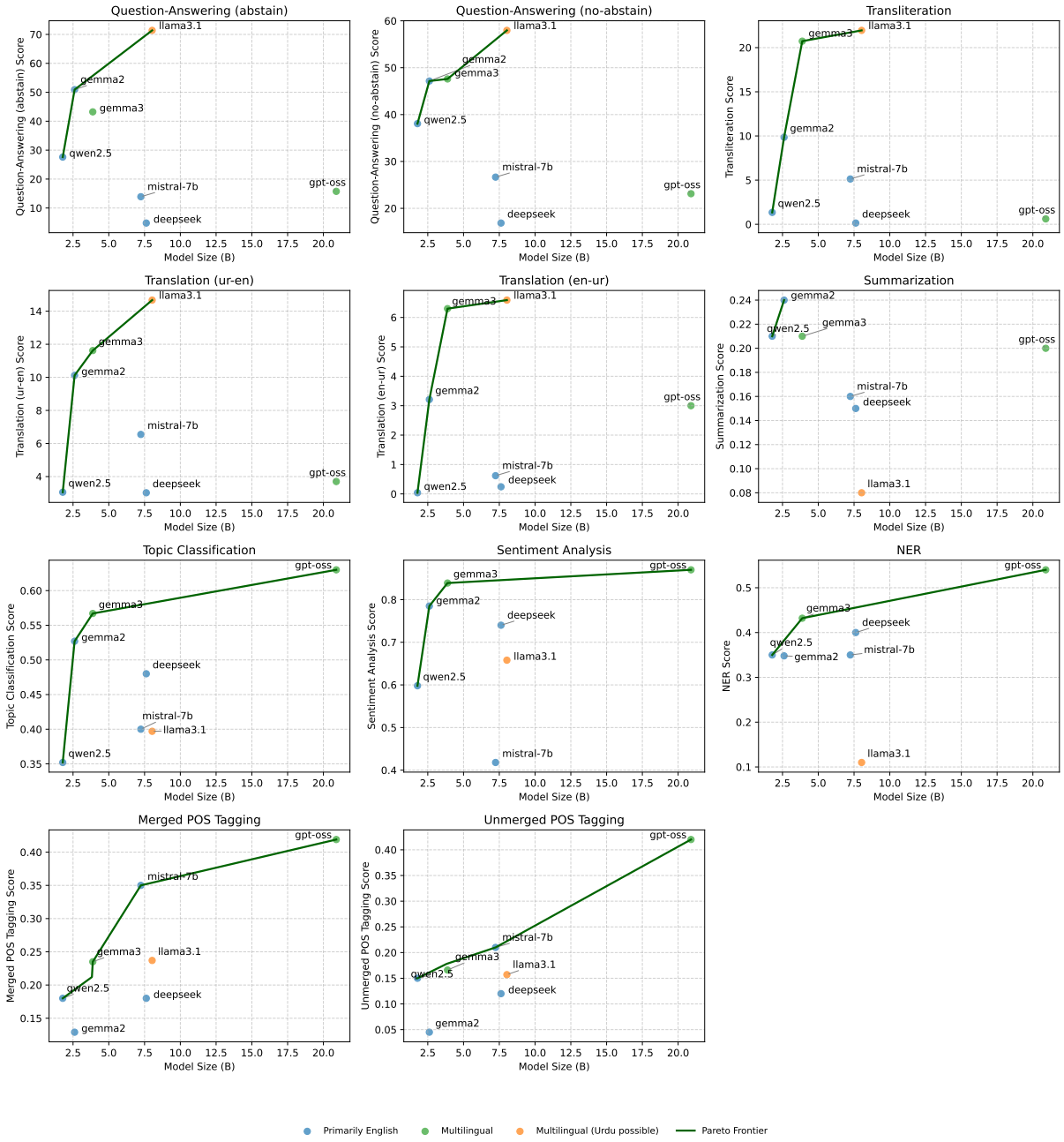

Figure 8: Pareto Frontiers for each task.

were more “natural”, “smooth”, and “accessible”.

Therefore, evaluating predicted translations against archaic and scriptural style translations without prompting the model to answer in that tone can be a misleading measure of model performance.

A second limitation of the SacreBLEU score is that annotations (feature (ii)) disrupt n-grams matching when they are present in only one of the two reference texts, lowering the score.

Among the alternatives, the COMET score can resolve these two issues by considering semantic overlap between the reference and prediction, whereas the chrF++ score faces the same limitations as it also based on character matching.

Lastly, dataset features (iii), (iv) and (v) lead to low scores regardless of the scoring scheme and model output.

B.2 Transliteration

Dataset. While the transliteration dataset itself does not contain any notable discrepancies, it features certain variations that are typical of Urdu transliteration. These variations also exist in model predictions, which can result in false mismatches while scoring.

As observed in the dataset and model predictions, Urdu is commonly transliterated (romanized) in multiple ways due to the following factors:

(i) Some Urdu sounds can be mapped to multiple English alphabets. For example, “ی” at the end of a word can be any of “i”, “y”, or “ee”.

(ii) There are multiple valid positions of spaces in transliterations, as words like reduplications commonly spoken as one unit can be joined in writing. For example, “lagbhag” and “lag bhag” can both be accurate, similar to “chitchat” and “chit-chat”.

(iii) Diaritic marks can be placed in Urdu transliterations to enhance pronunciation (e.g. “ā”) or represent special sounds (e.g an apostrophe for “ع”), while transliterations without them are readable and valid as well.

(iv) The Urdu script also contains important diaritic marks which are transliterated as connectors that can also be represented in multiple ways. For example, “wazeer-e-azam” or “wazeere azam”.

(v) English words, mainly nouns, can be written in their English spellings in the transliteration or in a more pronunciation-like manner, e.g. “Pennsylvania” or “pensalwania”.

Neither the dataset nor the model predictions follow one variation for each of these cases consistenly, reflective of real world Urdu romanizations as well.

Scoring. Using SacreBLEU to evaluate Urdu transliteration thus leads to low scores in the very common scenario of matching n-grams being disrupted by differences in only one letter, space, connector or diartic mark - despite the predicted transliterations being valid and faithful. The same limitation also persists with chrF++, as the differences are character-level.

B.3 Named Entity Recognition (NER)

Dataset. The dataset contains a class imbalance, with the ‘other’ class containing only 28 samples while all other classes contain >113 samples. The breakdown is shown in Table 4.

Label	Count
Time	143
Person	143
Organization	143
Other	28
Number	143
Location	143
Designation	114
Date	143

Table 4: NER dataset composition

Scoring. Using Macro-F_1 on this dataset without weighting for examples presents an unfaithful average. In our case, excluding the ‘other’ class from the calculation increases the score of our best model Gemma-3B by 0.05. The impact was not strong in this scenario because the model performed more poorly on the ‘time’ class, which also reduced the average significantly. In other scenarios, however, this can have a larger impact.

B.4 Part-of-Speech (PoS) Tagging

Dataset. The dataset is highly imbalanced, with the composition shown in Table 5. Interestingly, the most frequent model output with unmerged labels was ‘_’, which has 0 samples in the dataset, along with other labels inlcuding Symbol. Merging the labels reduces the total number of labels, but imbalances the data more, as shown in Figure 9.

Case	English	Urdu
(i)	And formed thee for Myself.	اور میں نے تجھے خاص اپنی ذات کے لئے پسند فرما لیا
(ii)	”blazing a trail of dust,”	پھر وہ اس (حملے والی) جگہ سے گرد و غبار اڑاتے ہیں،
(iii)	And all who are on the earth; so that this might deliver him.	اور جتنے زمین میں ہیں سب پھر یہ بدلہ دنیا اسے بچا لے
(iv)	San Benito. I can’t talk to you. Pair beats ace.	برائے مہربانی۔ پلیز۔ [وکٹر] پیر بند تھا۔ بجنا شروع ہو گیا۔ میں آگیا۔
(v)	”- I don’t know, what... - I know what you’re doing, - ahh, I don’t know..”	" ِ ""× ¾ ¾ ً ""¹ °×³",

Table 3: Examples of cases from the translation dataset

Scoring. In the original dataset, some classes have as low as 0 examples which highly inflates the impact of the model’s performance on those classes in the score. Similar to NER, using Macro-F_1 on this dataset without weighting for examples presents an unfaithful average.

Label	Before Merging	After Merging
Noun	127	252
Adposition	82	82
Number	79	79
Adjective	127	127
Particle	45	45
Determiner	37	37
Pronoun	59	59
Other	2	2
Adverb	74	74
Interjection	1	1
Verb	127	210
Symbol	0	10
Punctuation mark	10	0
Coordinating conjunction	8	0
Subordinating conjunction	14	0
Conjunction	0	22
Proper noun	125	0
Auxliary verb	83	0
_	0	0

Table 5: POS-tagging label counts before and after merging

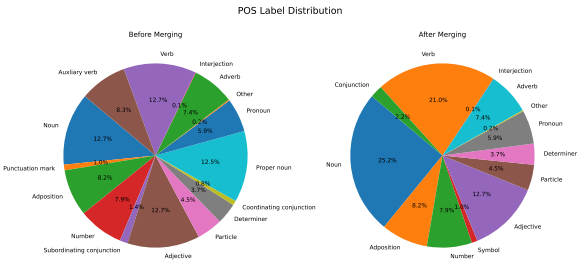

Figure 9: POS-tagging labels distribution before and after merging.

B.5 Benchmarking Recommendations

Based on our benchmark quality analysis, we propose a set of actionable suggestions to move beyond the current flawed evaluation paradigms and foster a more equitable ecosystem for low-resource languages. The central theme is a shift from purely automated, n-gram-based metrics to a more holistic, human-centric, and semantically-aware approach.

1. **Prioritize Human-Centric Dataset Curation:** The core issue stems from the unfaithful nature of many existing benchmark datasets, which are often built from poor automated translations or noisy web scrapes. We advocate for the creation and validation of datasets by expert linguists and native speakers. This process should also guide the curation of synthetic data, ensuring that even machine-generated content is vetted for cultural and linguistic accuracy.

2. **Adopt Semantic and Human-Preference Scoring:** Our findings show that metrics like SacreBLEU are misleading, as they rely on exact surface-level matches and penalize valid, natural-sounding outputs that differ stylistically from an archaic reference text. We suggest prioritizing more semantic scoring schemes like COMET, which can resolve issues with both stylistic differences and annotation mismatches by evaluating semantic overlap. As a scalable alternative, employing LLMs-as-a-judge can effectively evaluate subjective qualities like naturalness and accessibility, as demonstrated in our experiment with ChatGPT. To create a gold-standard benchmark, the community should invest in building human-preference evaluation platforms for low-resource languages, creating an ”Urdu Arena” to capture what real users find most helpful.

3. **Invest in Language-Specific Tooling and Benchmarks:** The ”one-size-fits-all” approach to LLM development and evaluation fails low-resource languages. When building or quantizing models, evaluation should be performed on language-specific benchmarks rather than relying solely on generalist multilingual test sets. Furthermore, developing better, language-specific tokenizers is crucial. Current tokenizers often fail to properly handle the morphological complexity of languages

You are a [ROLE]. [INSTRUCTION]. Only return the [TARGET] without any additional text or explanations.
[TEXT TYPE]:
[INPUT TEXT]

Figure 10: Evaluation Prompt Template

like Urdu (e.g., diacritics, connectors), leading to inefficient representations and downstream performance degradation.

C Evaluation Prompts

Figure 10. In the template, the following placeholders are used:

- ROLE: the persona of the LLM e.g. Urdu topic classifier, English to Urdu translator.
- INSTRUCTION: A brief command specifying the precise instruction for the given task.
- TARGET: the precise output required e.g. topic or sentiment label.
- INPUT TYPE: the precise input text being provided e.g. news or movie review.
- INPUT TEXT: the input text for the task.

Important exceptions to the template are question-answering and transliteration. We do not provide the ROLE for question-answering as it only increases the prompt length without adding meaningful specificity to the model's default behaviour. For transliteration, we also add the definition of the transliteration task before the INSTRUCTION, as some models perform translations otherwise. Lastly, summarization is evaluated in a few-shot setting.

For some tasks, additional instructions are incorporated to align responses with the gold text format, enabling authentic scores. For example, the transliteration prompt specifies "do not add diacritic marks" to match the dataset, as the marks' inclusion - though correct - yields low scores.

Question Answering "Answer the given question using only the provided context. The answer must be in Urdu and should be a short phrase, not a full sentence. If the answer is not in the context, respond with 'جواب موجود نہیں'."

Transliteration "Transliteration is to convert text from one script to another keeping the pronounciation/sounds the same. Transliterate the given Pakistani Urdu text into English script. Do not add any diacritic marks. Only return the transliteration without any explanations, notes or extra text."

Translation (en-ur) "You are an English to Urdu translator. Translate the provided English text to Urdu. Only return the translation without any additional text or explanations."

Translation (ur-en) "You are an Urdu to English translator. Translate the provided Urdu text to English. Only return the translation without any additional text or explanations."

Sentiment Analysis "You are an Urdu sentiment classifier. Classify the given movie review as 'positive' or 'negative' based on the sentiment. Only return the sentiment label without any explanations or extra text."

Topic Classification "You are an Urdu topic classifier. Classify the given Urdu news text into one of the following topics: 'business', 'entertainment', 'health', 'politics', 'science', 'sports', 'world', 'other'. Only return the topic without any explanations or extra text."

Named Entity Recognition (NER) "You are an Urdu named entity recognizer. Read the sentence below, and classify the word wrapped in the <h1> tag into one of the following labels based on its context: 'person', 'organization', 'number', 'location', 'designation', 'date', 'time', 'other'. Only return the label without any explanations or extra text."

Part-of-Speech Tagging (Unmerged) "You are an Urdu part-of-speech tagger. Read the sentence below, and classify the word in the <h1> tag into one of the following part-of-speech labels: 'noun', 'punctuation mark', 'adposition', 'number', 'symbol', 'subordinating conjunction', 'adjective', 'particle', 'determiner', 'coordinating conjunction', 'proper noun', 'pronoun', 'other', '_', 'adverb', 'interjection', 'verb', 'auxiliary verb'. Only return the label and no other text."

Part-of-Speech Tagging (Merged) "You are an Urdu part-of-speech tagger. Read the sentence below, and classify the word in the <h1> tag into one of the following part-of-speech labels: 'noun', 'verb', 'adjective', 'adverb', 'pronoun', 'particle',

Model	Hugging Face Model Card
Qwen 2.5	Qwen/Qwen2.5-1.5B-Instruct-GGUF
DeepSeek-R1-Distill-Qwen	bartowski/DeepSeek-R1-Distill-Qwen-7B-GGUF
LLaMA 3.1	bartowski/Meta-Llama-3.1-8B-Instruct-GGUF
Gemma 2B	bartowski/gemma-2-2b-it-GGUF
Gemma 3B	ggml-org/gemma-3-4b-it-GGUF
Phi-3.5 Mini	bartowski/Phi-3.5-mini-instruct-GGUF
Mistral-7B	TheBloke/Mistral-7B-v0.1-GGUF
GPT-OSS	unsloth/gpt-oss-20b-GGUF

Table 6: Evaluated Model Cards

'conjunction', 'interjection', 'determiner', 'adposition', 'number', 'symbol', 'other'. Only return the label and no other text."

D Computing Infrastructure

We ran evaluations on a virtual machine with NVIDIA GeForce RTX 3060 12GB. The total GPU hours were 20 hours including reruns.

E Model Cards

We provide the official model cards for all models evaluated in this study in Table 6.

F Datasets

F.1 Licenses

The Dakshina dataset (Roark et al., 2020) for transliteration is under CC BY-SA 4.0, and OPUS-100 (Zhang et al., 2020; Tiedemann, 2012) for translation is released under the GPL 3.0 license. The XSUMUrdu (Munaf et al., 2024) summarization dataset is released under CC BY 4.0. The UQuAD dataset is under CC0-1.0, while UQA (Arif et al., 2024b) is licensed under CC BY 4.0. The Urdu IMDB sentiment analysis dataset uses the OBDL license. The Urdu NER dataset is available under the MIT license. The Urdu-News-Headline dataset for Topic Classification along with Universal Dependencies for POS-tagging (Nivre et al., 2020) is provided under CC BY 4.0.

All datasets used in this study are fully compliant with their respective licenses.

F.2 Size

Sizes of each dataset are given in Table 7.

Task	Dataset Size
Question Answering	1000
Summarization	1000
Transliteration	1000
Translation	1000
Sentiment Analysis	1000
Part-of-Speech (PoS) Tagging	1000
Named Entity Recognition	1000
Topic Classification	1000

Table 7: Dataset Sizes for Evaluated Tasks