LOLA – An Open-Source Massively Multilingual Large Language Model

Nikit Srivastava[1], **Denis Kuchelev**, **Tatiana Moteu Ngoli**[1], **Kshitij Shetty**[2], **Michael Röder**[1], **Hamada M. Zahera**[1], **Diego Moussallem**[1], **Axel-Cyrille Ngonga Ngomo**[1]

Data Science Group, Paderborn University, Germany

[1] {nikit.srivastava, tatiana.moteu, michael.roeder, hamada.zahera, diego.moussallem, axel.ngonga}@upb.de
[2] kshitij@mail.upb.de

Abstract

This paper presents LOLA, a massively multilingual large language model trained on more than 160 languages using a sparse Mixture-of-Experts Transformer architecture. Our architectural and implementation choices address the challenge of harnessing linguistic diversity while maintaining efficiency and avoiding the common pitfalls of multilinguality. Our analysis of the evaluation results shows competitive performance in natural language generation and understanding tasks. Additionally, we demonstrate how the learned expert-routing mechanism exploits implicit phylogenetic linguistic patterns to potentially alleviate the curse of multilinguality. We provide an in-depth look at the training process, an analysis of the datasets, and a balanced exploration of the model's strengths and limitations. As an open-source model, LOLA promotes reproducibility and serves as a robust foundation for future research. Our findings enable the development of compute-efficient multilingual models with strong, scalable performance across languages.

1 Introduction

Large Language Models (LLMs) have shown tremendous capability across a diverse set of tasks in recent years (Radford et al., 2019; Kaddour et al., 2023). This progress has propelled research, with many chat-based LLMs[1] gaining popularity among general users. However, concerns remain, particularly regarding their accessibility for multilingual usage (Joshi et al., 2020) and open-source licensing policies (Liu et al., 2023). The number of competent LLMs significantly decreases for languages other than English (Üstün et al., 2024). This, combined with *the curse of multilinguality*—a phenomenon in which the ability of models to generalize across multiple languages diminishes unless their capacity is significantly expanded (Conneau et al., 2020)—means non-English speakers often have access to inferior systems. Additionally, many new models (Jiang et al., 2023; Achiam et al., 2024; Dubey et al., 2024) are pay-to-use, require personal information, or do not fully disclose training details, creating significant hurdles for multilingual research.

To advance multilingual language modeling, we introduce LOLA,[2] a massively multilingual model that follows a GPT-style (Radford et al., 2019) decoder-only architecture with sparse Mixture-of-Experts (MoE) layers (Shazeer et al., 2017). MoE architectures have shown strong performance on learning the underlying structure of the data (Chen et al., 2022) but their application in multilingual LLMs remains underexplored. MoE models can effectively increase model capacity with minimal additional computational cost, offering the possibility of leveraging implicit clusters like language family groups and playing a crucial role in addressing the challenges of multilinguality.

Language family groups, consisting of languages sharing common ancestral roots, offer opportunities for enhancing language models. Despite linguistic diversity, these families exhibit structural, syntactic, and semantic similarities (Rowe and Levine, 2015) that can be exploited to improve performance across related languages. Our goal is to leverage MoE's strengths to exploit the phylogenetic structure of languages and achieve better prediction performance. In particular, the shared and non-shared parameters of MoE-based models offer a promising approach to mitigating *the curse of multilinguality* by increasing capacity while remaining compute efficient (Shazeer et al., 2017). By exploiting language families, we aim to

[1] ChatGPT: chat.openai.com;
LLAMA: llama.meta.com;
Mistral: mistral.ai;
Gemini: gemini.google.com;
Deepseek: deepseek.com.

[2] Source Code: github.com/dice-group/LOLA;
Model Weights: huggingface.co/dice-research/lola_v1.

Proceedings of the 31st International Conference on Computational Linguistics, pages 6420–6446
January 19–24, 2025. ©2025 Association for Computational Linguistics

close gaps in current models—particularly for low-resource languages—by enhancing cross-linguistic transfer learning.

Another important factor is the availability of LLMs as a free resource, accessible for anyone to use, modify, and redistribute without discrimination against any individuals or purposes. Many popular LLMs that claim to be "open source" either withhold their training datasets (e.g., Mistral, Grok[3]), fail to publish their training code (e.g., Llama, Grok), or do not release their inference code (e.g., Grok-2[4]) (Spectrum, 2024). In some cases, these models are released under licenses that are restrictive, discriminatory, or impose additional conditions (Liesenfeld et al., 2023; Liesenfeld and Dingemanse, 2024). The artifacts and components used in LOLA were selected based on their suitability for training massively multilingual LLMs while minimizing licensing concerns. All chosen components are obtainable, modifiable, and redistributable in accordance with the terms of their original licenses.

To assess LOLA's performance, we evaluated it on four task types: 1. Question Answering (Q&A), 2. Reasoning, 3. Natural Language Inference (NLI), and 4. Reading Comprehension. In total, we assessed the model across 13 multilingual tasks, comparing it to 17 other models grouped into three categories based on their active parameter count.[5] Our results demonstrate strong performance across most tasks, though we note the limitations in 1. tasks involving factual and mathematical Q&A; and 2. comparisons with models that use more than five times the active parameters of LOLA. These findings are discussed in detail later in the paper.

Beyond presenting the multilingual model as our main contribution, we address the following key research questions:

1. *Does training a Mixture-of-Experts model on a wide variety of languages enhance generalization or lead to confusion?*

2. *How do experts impact the model's capacity to leverage implicit language groups?*

3. *What are the potential limitations?*

[3] github.com/xai-org/grok-1

[4] x.ai/blog/grok-2

[5] The number of parameters a model utilizes per token (Fedus et al., 2022). This distinction is crucial for understanding the efficiency and performance of MoE models.

2 Related Work

The development of LLMs has gained significant momentum since the introduction of the Transformer architecture by Vaswani et al. (2017). As LLMs grew in size and complexity, their capacity to model increasingly nuanced linguistic patterns expanded. Models like GPT3 and Llama (Brown et al., 2020; Touvron et al., 2023) showcased the ability of large models to perform few-shot learning, a significant milestone that further highlighted the flexibility of Transformer-based architectures. As the need to extend their capabilities to handle multiple languages effectively became increasingly apparent, research into multilingual LLMs surged, aiming to enable performance across diverse languages with a single model, reducing the need for language-specific systems (Zhu et al., 2024). Key efforts in this area include systems such as mBERT, XLM-R, mT5, and BLOOM (Devlin et al., 2019; Conneau et al., 2020; Xue et al., 2021; Scao et al., 2022), with more recent models like Tower, SeaLLM, and Breeze (Alves et al., 2024; Nguyen et al., 2024b; Hsu et al., 2024) focusing on adapting primarily English-pretrained models into multilingual ones through continued training. However, research in multilingual LLMs faces several challenges, particularly in balancing performance across languages while keeping training costs manageable, as emphasized by Conneau et al. (2020).

One of the significant challenges with scaling LLMs is the computational cost associated with training and deploying models with billions or trillions of parameters. To address this, the Mixture-of-Experts (MoE) paradigm has emerged as a promising approach for efficiently scaling large models. The MoE architecture proposed by Shazeer et al. (2017) introduces the concept of sparsity, where only a subset of the model's parameters is activated during each forward pass, thereby reducing the computational burden while maintaining high performance. Their approach demonstrated that models could achieve state-of-the-art performance while being computationally efficient. Later approaches, such as GShard and Switch Transformers (Lepikhin et al., 2021; Fedus et al., 2022), extended the MoE framework by simplifying routing and enhancing scalability, enabling models with over a trillion parameters while maintaining efficient computational costs and setting new benchmarks in large-scale model training. These advances led to increased research in MoE-based LLMs, resulting in models like GLaM, DeepSpeed MoE and Mixtral (Du et al., 2022; Rajbhandari et al., 2022; Jiang et al., 2024).

Given the unique architecture of the MoE-based LLMs, Machine Translation (MT) models have explored its potential in language grouping. Several MT systems, such as M2M, NLLB, and Lingual-SMoE (Fan et al., 2021; Team et al., 2022; Zhao et al., 2024), have trained MoE-based models to enable many-to-many translation, leveraging either learned or custom expert-routing mechanisms that assigns experts based on the language. Systems like NLLB continue to demonstrate state-of-the-art MT performance to this day (Zhu et al., 2024). In the case of pre-trained base models, Zoph et al. (2022) briefly touch upon the multilingual nature of MoE models, though they primarily note that expert load balancing loss constrains the model's capacity to assign language-specific experts. Despite these advances, the application of MoE for pre-training massively multilingual LLMs remains underexplored. This research contributes to addressing that gap.

3 Model Overview

Our model is based on a GPT-style (Radford et al., 2019) decoder-only Transformer architecture (Vaswani et al., 2017). We replace the standard feed-forward layers (FFNs) with Mixture-of-Experts (MoE) layers in every alternate Transformer layer. These MoE layers utilize a *top-1 gating* mechanism inspired by the Switch Transformer (Fedus et al., 2022) due to its simplicity and effectiveness. The architecture consists of 24 decoder layers with a model hidden and embedding dimension of 2048, 16 attention heads, a maximum sequence length of 2048, and each MoE layer includes 16 experts. We use the GELU (Hendrycks and Gimpel, 2017) non-linearities and the Adam (Kingma and Ba, 2015) optimizer for our model. Based on this configuration, our model has 1.3 billion active parameters out of 7.4 billion total parameters. Due to this sparsity, our model has a training/inference cost similar to that of a 1.3 billion dense model.[6] Figure 1 provides a multi-level overview of the model architecture. The model configuration and training are facilitated using the Megatron-DeepSpeed[7] framework, which is based on Shoeybi et al. (2020); Rajbhandari et al. (2022).

[6]Number of parameters activated in a single forward and backward pass.

[7]github.com/microsoft/Megatron-DeepSpeed

3.1 Routing Mechanism in MoE Layers

For routing tokens through the MoE layers with N (i.e., 16) experts, we first compute the logits for the gating function. These logits are then passed through a *Softmax* function to calculate the probability for each expert:

$$h(x) = W_g \cdot x, \tag{1}$$

$$G_i(x) = \frac{\exp(h(x)_i)}{\sum_{j=1}^{N} \exp(h(x)_j)}, \tag{2}$$

where $h(x)$ contains the logit vectors for all experts, W_g is the gating weight matrix, and x is the input. The logit vector and gating probability of the i-th expert is denoted by $h(x)_i$ and $G_i(x)$ respectively.

Once the gating probabilities are computed, the output of the MoE layer is calculated by selecting the most probable expert i^* and multiplying its gating probability $G_{i^*}(x)$ with the output of the corresponding expert $E_{i^*}(x)$:

$$i^* = \arg\max_i G_i(x), \tag{3}$$

$$\text{MoE}(x) = G_{i^*}(x) \cdot E_{i^*}(x). \tag{4}$$

3.2 Training and Loss Functions

Our model is pre-trained using a causal language modeling task (Radford and Narasimhan, 2018), where the objective is to minimize the cross-entropy loss alongside an auxiliary MoE loss.This auxiliary loss, inspired by works such as Shazeer et al. (2017), Lepikhin et al. (2021), and Fedus et al. (2022), is used to ensure stable training and effective load balancing among the experts. The auxiliary loss incorporates two vectors:

- P represents the average weight assigned to all tokens for each expert.
- f denotes the fraction of tokens allocated to each expert.

Given an input sequence $S = \{s_1, s_2, s_3, \ldots, s_T\}$ of length T, and N experts in each MoE layer, for each expert $i = 1, 2, \ldots, N$, these vectors are defined as:

$$P_i = \frac{1}{T} \cdot \sum_{t=1}^{T} G_i(s_t), \tag{5}$$

$$f_i = \frac{1}{T} \cdot \sum_{t=1}^{T} M_i(s_t), \tag{6}$$

where:

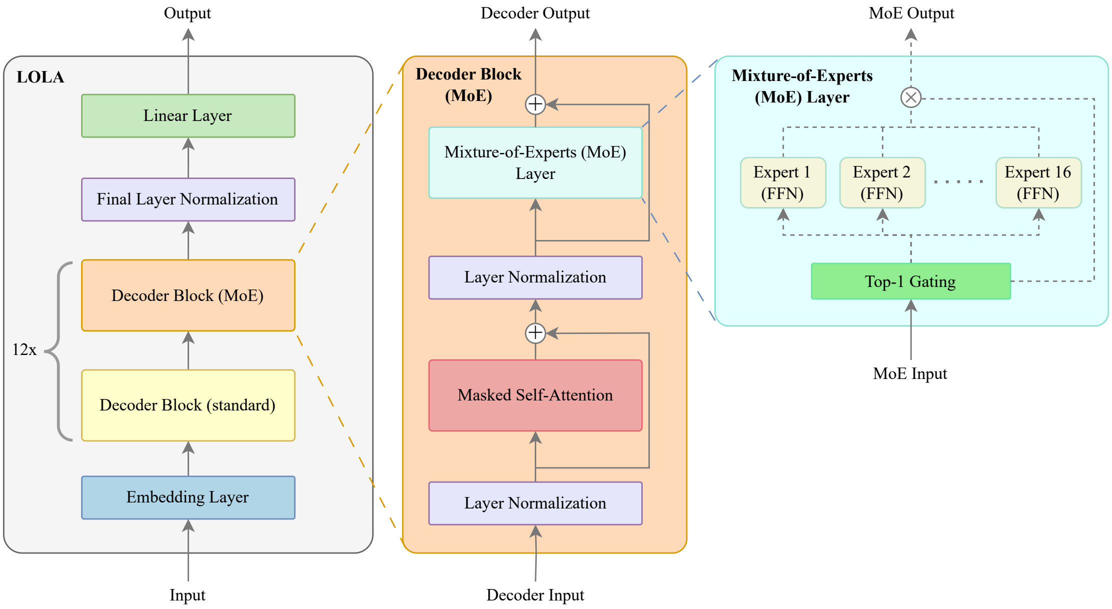

Figure 1: Three-level overview of the LOLA architecture. The left-most block provides a high-level overview of the layers within LOLA, including the alternating standard and Mixture-of-Experts (MoE)-based decoder blocks. The middle block gives a detailed view of the MoE-based decoder block structure. The right-most block zooms in on the inner workings of each MoE layer, showing how the top-1 gating mechanism selects from multiple expert Feed Forward Networks (FFNs).

- $G_i(s_t)$ is the gating weight assigned to expert i for token s_t,
- $M_i(s_t)$ is a binary mask indicating whether token s_t is routed to expert i, determined by the top-1 gating mechanism (Shazeer et al., 2017).

The auxiliary loss l_{aux} is formulated as:

$$l_{\text{aux}} = N \cdot \sum_{i=1}^{N} P_i \cdot f_i, \quad (7)$$

which represents the scaled dot product between P and f.

For the language modeling task, the cross-entropy loss is computed as:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{T} \cdot \sum_{t=1}^{T} \log p(s_t \mid s_{<t}). \quad (8)$$

The final loss function for the model combines the cross-entropy loss and the auxiliary loss:

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{CE}} + \alpha \cdot l_{\text{aux}}, \quad (9)$$

where α is the multiplicative coefficient for the auxiliary loss. Throughout this work, we set $\alpha = 10^{-2}$ based on the recommendations by Fedus et al. (2022).

3.3 Training Data and Setup

The model was trained on data sampled from the CulturaX (Nguyen et al., 2024a) dataset, which consists of raw text documents in 167 languages, amounting to over 6 trillion tokens from more than 7 billion documents (see Appendix A.6 for train sample details). We tokenized the data using the SentencePiece (Kudo and Richardson, 2018) tokenizer with a vocabulary size of 100,000.

Training was conducted on 96 NVIDIA A100 GPUs[8] with a total compute of approximately 44,000 GPU hours. The model was trained for 19 days, consuming a total of 465 billion tokens across a batch size of 768 documents.[9]

4 Evaluation

4.1 Models

After reviewing the available multilingual LLMs, we selected 17 models with active parameters ranging from 300 million to 7.5 billion. Table 1 provides a list of the selected models along with further details. The selection was based on the following criteria: 1. They are base pretrained models without

[8] GPU Model: NVIDIA A100-SXM4-40GB
[9] Further training details in Appendix A.2

Model	Params (B)	Consumed Tokens (T)	Max Seq. Length	Languages	Category
Glot500m (Imani et al., 2023)	0.39	-	512	500	1
XLM-R Large (Conneau et al., 2020)	0.55	6	512	100	1
mBART (Liu et al., 2020)	0.68	1.8	1024	25	1
BLOOM-1B1 (Scao et al., 2022)	1.10	0.341	Arbitrary	48	1
MT5 Large (Xue et al., 2021)	1.20	1	Arbitrary	101	1
mGPT (Shliazhko et al., 2024)	1.30	0.440	2048	61	1
BLOOM-1B7 (Scao et al., 2022)	1.70	0.341	Arbitrary	48	1
XLM-R XL (Conneau et al., 2020)	3.50	6	Arbitrary	100	2
MT5 XL (Xue et al., 2021)	3.70	1	Arbitrary	101	2
UMT5 XL (Chung et al., 2023)	3.70	1	Arbitrary	107	2
TowerBase 7B (Alves et al., 2024)	6.74	2	Arbitrary	10	3
Mistral v0.3 (Jiang et al., 2023)	7.00	-	32768	5	3
Falcon (Almazrouei et al., 2023)	7.00	1.5	2048	2	3
BLOOM-7B1 (Scao et al., 2022)	7.10	0.366	Arbitrary	48	3
SeaLLM v2 (Nguyen et al., 2024b)	7.38	-	Arbitrary	10	3
SeaLLM v2.5 (Nguyen et al., 2024b)	7.38	-	Arbitrary	10	3
Breeze (Hsu et al., 2024)	7.49	-	Arbitrary	2	3
LOLA (Our Model)	1.3	0.465	2048	167	1

Table 1: Characteristics of models used for comparison in the evaluation, including model names, active parameter sizes (in billions), the number of consumed tokens (in trillions), maximum sequence length, and the number of languages each model was trained on. The models are grouped by their size categories (see appendix Figure 4).

any fine-tuning; 2. The weights are openly accessible without requiring personal information beyond name and email; 3. Model weights are available via Huggingface[10] [11]; 4. The models are compatible with our evaluation hardware setup.[12] Given the wide range of active parameters, we decided to group the models based on their sizes. We employ the *distortion*[13] and *silhouette*[14] scores to determine the optimal number of categories, which was identified as 3 (see Appendix A.1). Subsequently, K-Means clustering was used to classify the models into 3 categories (1-3). Although LOLA falls within *Category-1*, we compare and analyze its performance against each category.

4.2 Tasks

We evaluate LOLA on 13 multilingual benchmarks datasets/tasks: *ARC* (Clark et al., 2018), *HellaSwag* (Zellers et al., 2019), *LAMBADA* (Paperno et al., 2016), *MMLU* (Hendrycks et al., 2021), *MGSM Direct and MGSM Native CoT* (Shi et al., 2022), *PAWS-X* (Yang et al., 2019), *TruthfulQA* (Lin et al., 2022a), *XCOPA* (Ponti et al., 2020), *XNLI* (Conneau et al., 2018), *XStoryCloze* (Lin et al., 2022b), *XWinograd* (Tikhonov and Ryabinin, 2021), and *Belebele* (Bandarkar et al., 2023). We use the multilingual versions of originally English tasks (*ARC*, *HellaSwag*, *MMLU*, and *TruthfulQA*) introduced in *OKAPI* by Dac Lai et al. (2023). Details of these evaluation tasks are provided in Table ??. We utilize the *Language Model Evaluation Harness* framework by Gao et al. (2024) for evaluations. Examples from these tasks can be found in Appendix A.3.

Type	Task	Languages
Q&A	ARC	31
	MGSM (Direct)	11
	MGSM (Native CoT)	11
	TruthfulQA	31
	MMLU	34
Reasoning	HellaSwag	30
	XCOPA	11
	XStoryCloze	11
	XWinograd	6
NLI	PAWS-X	7
	XNLI	15
Reading Comprehension	LAMBADA	5
	Belebele	122

Table 2: Evaluation tasks used to evaluate LOLA, along with the number of languages covered by each task.

We group the tasks into four main categories: 1. Question Answering (Q&A) 2. Reasoning, 3. Natural Language Inference (NLI), and 4. Read-

[10] huggingface.co

[11] Required for the evaluation framework.

[12] Single NVIDIA A100 with 40GB GPU memory, 100GB of CPU memory, and 16 CPU cores.

[13] Mean sum of squared distances to centers.

[14] Mean ratio of intra-cluster and nearest-cluster.

ing Comprehension. We briefly describe each category and the corresponding tasks below:

4.2.1 Question Answering (Q&A)

This category includes tasks that require knowledge across various domains such as mathematics, philosophy, law, and medicine. *ARC* is a multiple-choice science question dataset for grades 3 to 9, requiring reasoning (Clark et al., 2018). *MGSM* is a benchmark of grade-school math problems requiring multi-step reasoning, with two variations: *MGSM (Direct)* and *MGSM (Native CoT)*, the latter including Chain-of-Thought prompts in the target language[15] (Shi et al., 2022). *TruthfulQA* measures a model's ability to generate truthful answers to factual questions (Lin et al., 2022a). *MMLU* is a large-scale multitask benchmark of multiple-choice questions spanning a wide range of topics (Hendrycks et al., 2021).

4.2.2 Reasoning

This category includes tasks that require commonsense reasoning. *HellaSwag* assesses a model's commonsense reasoning capabilities (Zellers et al., 2019). *XCOPA* evaluates a model's ability to transfer commonsense reasoning across multiple languages (Ponti et al., 2020). *XStoryCloze* tests understanding of everyday situations through causal and relational information in daily events (Lin et al., 2022b). *XWinograd* is a multilingual version of the Winograd Schema Challenge, requiring resolution of ambiguities in sentences differing by only one or two words, necessitating world knowledge and complex reasoning (Tikhonov and Ryabinin, 2021).

4.2.3 Natural Language Inference (NLI)

This category assesses the ability to identify relationships between sentences, such as paraphrasing and textual entailment. *PAWS-X* contains challenging paraphrase identification pairs derived from Wikipedia and Quora (Yang et al., 2019). *XNLI* evaluates cross-lingual sentence representations by testing textual entailment (Conneau et al., 2018).

4.2.4 Reading Comprehension

This category assesses reading comprehension abilities, requiring models to predict the next word or select the correct answer from given options. *LAMBADA* evaluates a model's text understanding through word prediction (Paperno et al., 2016). *Belebele* is a multilingual reading comprehension dataset evaluating models on languages with varying resource levels (high, medium, and low) (Bandarkar et al., 2023).

4.3 Performance Metrics

As evaluation metrics, we employ the following:

Accuracy is a metric that assesses how frequently an input is predicted by the model to be the correct class. It is calculated by computing the ratio of correctly predicted instances to the total number of instances. This metric is used by all evaluation tasks except *MGSM*.

Exact Match measures the match between a reference and predicted parameter. It sums the exact match scores (1 for an exact match, 0 otherwise) and divides by the total number of predictions. This metric is used only for *MGSM* tasks, utilizing the *flexible-extract* implementation by Gao et al. (2024) to account for formatting differences.

4.4 Results

We configure our experiments based on each distinct combination of task, model, language, and the number of shots for few-shot learning. The shot settings include zero-shot, one-shot, and few-shot (i.e., 5). Altogether, we perform over 14,000 unique experiments. Given the extensive scale of these experiments, the results are not included directly in the main text for brevity. Instead, information and links to the detailed result tables are provided in Appendix A.4. A comprehensive analysis and discussion over these results is presented in the subsequent section.

5 Analysis

We present our analysis of LOLA in two subsections. In the first subsection, we discuss our key insights derived from the evaluation results. Next, we analyze LOLA's learned MoE routing, focusing on its ability to leverage language family groupings, which aligns with our core motivation and intuition behind MoE for multilingual LLMs.

5.1 Result Analysis

We assess LOLA's performance relative to other models by evaluating the results across all languages for each task, employing two methods: 1. using the Wilcoxon signed-rank test (Wilcoxon, 1945) to determine the statistical significance of differences between performance distributions (with

[15]The target language for model evaluation.

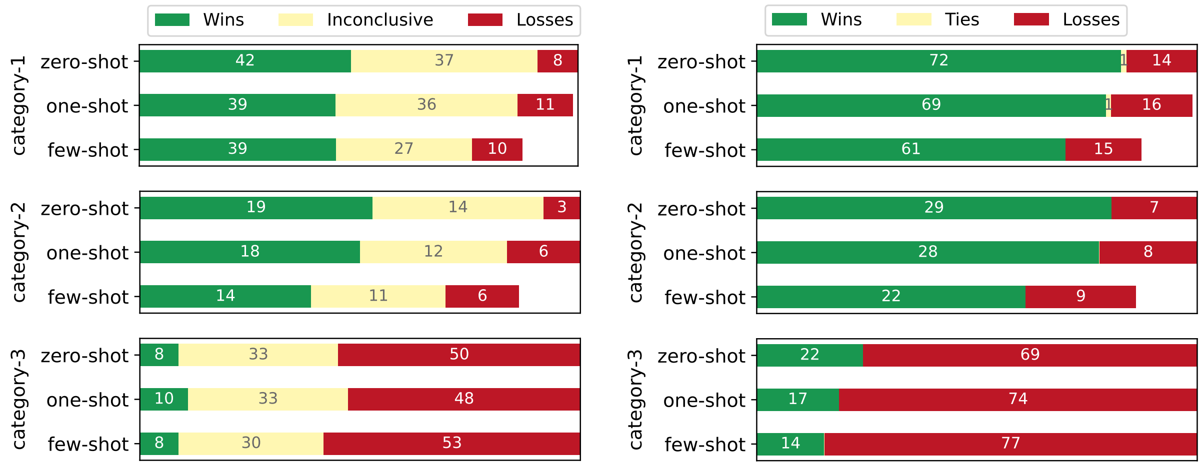

Figure 2: Comparison of LOLA's zero-, one- and few-shot performance against the other multilingual models across all supported combinations of tasks and languages, categorized by model size. The left side shows the results from the Wilcoxon signed-rank test, indicating whether LOLA significantly outperforms (Wins), shows no significant difference (Inconclusive) or is outperformed by (Losses) other models. On the right is the average performance comparison to confirm whether LOLA is on average better than (Wins), the same as (Ties), or worse than (Losses) the other models.

a p-value threshold of 0.05); and 2. comparing average performance across all languages to provide a simplified overview.

These comparisons allow us to examine LOLA's performance across various levels of granularity, including: 1. the model's overall performance against all other models on the full set of tasks and languages; 2. its performance on specific task types; and 3. its performance on individual tasks. For brevity, we discuss the model's overall performance in this subsection, with more detailed analyses provided in Appendix A.5.

Figure 2 shows that LOLA consistently outperforms *Category-1* and *Category-2* models but underperforms relative to *Category-3* models, which are at least five times larger (see Table 1). Nonetheless, LOLA's strong performance against *Category-2* models—on average 2.8 times larger and trained on twice as many tokens—highlights its efficiency in multilingual settings with a substantially smaller computational footprint.

To summarize the finer granularity levels (Appendix A.5), we derive the following additional key insights about LOLA's performance:

Strengths: 1. strong performance in NLI, Reasoning, and Reading Comprehension tasks; and 2. competitiveness with *Category-3* models in NLI tasks.

Weaknesses: 1. limited gains on Q&A tasks, with particularly poor performance on *MGSM*; and 2. inferior few-shot performance compared to zero- and one-shot settings.

While the model's strengths can be attributed to its generalization capabilities, its weaknesses may be due to several factors. The subpar Q&A performance may stem from LOLA's limited factual grounding due to restricted training data per language (Fierro and Søgaard, 2022). Furthermore, the challenges on *MGSM* are likely due to the lack of a specialized tokenizer for arithmetic data and the absence of coding and LATEX data during training (Yuan et al., 2023). The diminished few-shot performance may be caused by the model's 2048-token sequence limit, which truncates essential context.[16]

These findings contribute to answering our first research question: *Does training a Mixture-of-Experts model on a wide variety of languages enhance generalization or lead to confusion?* The results indicate that training across diverse languages enhances generalization, particularly in NLI, Reasoning, and Reading Comprehension tasks; challenges persist in Q&A tasks, which may necessitate additional data or specialized pre-training.

5.2 MoE Analysis

In this subsection, we discuss our second research question: *How do experts impact the model's capacity to leverage implicit language groups?* We answer this question by analyzing whether there

[16] During evaluation overflowing sequences are truncated from the left.

is a correlation between the activity of the experts within the model and groups of languages that share common features. To this end, we measure the activation of the experts on all layers across 106 languages.[17] Based on these activities, we create a vector for each language comprising the activation of the experts when processing documents of this language. Based on these vectors, we calculate a language-to-language distance matrix using the normalized Euclidean distance. We compare our distance matrix with distance matrices of the URIEL project (Littell et al., 2017) comprising pairwise language distances based on a variety of features like 1. their syntactic features, 2. their phonological features, 3. their geographical location, and 4. their position in the Glottolog tree of language families (Hammarström et al., 2015). We calculate the Pearson correlation coefficients between these matrices and our matrix. Our results indicate a weak positive linear correlation between the activity of our model's experts and the distance of the languages within the language family tree. This correlation grows stronger when we focus the analysis on those languages for which the model saw more training documents, up to a correlation of 0.55 for the 23 languages that have at least 1 million documents in our training data.[18] For example, in our activity-based matrix, as well as in the family tree, Portuguese is closer to Spanish, French, Italian and Romanian than to the other 18 languages. Similarly, Swedish and Danish are very close to each other. This finding is in contrast to Zoph et al. (2022), who did not identify any specialization of experts in their model. However, for many family pairs, the tree-based distances are the maximum distance 1.0 because the languages are in different branches of the tree and do not share any common parent nodes. In our expert activity matrix, these values are typically lower. Therefore, while the experts seem to focus on certain languages, this focus is not very strict and they may still become active for other languages. A good example is the pairing of Arabic and Persian, which, despite belonging to different branches of the language family tree, exhibit a relatively small distance in the expert activity matrix. We provide more details of this analysis in Appendix A.6.

[17] Languages for which CulturaX has at least 10,000 documents.

[18] The Pearson correlation values for all 106 languages, the 93 languages with at least 10,000 training documents, and the 48 languages with at least 100,000 training documents are 0.27, 0.28, and 0.35, respectively. The 23 languages are ar, cs, da, de, el, en, es, fa, fi, fr, hu, it, ja, nl, pl, pt, ro, ru, sv, tr, uk, vi, and zh.

6 Discussion

LOLA demonstrates significant performance improvements over models with up to three times its active parameters. It effectively generalizes across a diverse range of languages, as observed in its performance on the *Belebele* benchmark, which includes 122 languages spanning both high- and low-resource categories (see Appendix A.5.4). This strong multilingual performance is achieved despite being trained on a relatively modest compute budget, showcasing its efficiency in large-scale language modeling. Our analysis reveals that the model successfully learns language groupings through expert routing, validating our initial intuition. This finding provides valuable insights, challenging previous assumptions about the MoE architecture's ability to capture language structures.

7 Conclusion

In this paper, we present LOLA, a compute-efficient, open-source multilingual language model. LOLA balances efficiency with increased capacity by utilizing a sparse MoE architecture, thus enabling effective generalization across diverse languages. Our model outperforms others in multilingual NLP tasks, even those with up to three times the active parameters. We also analyzed the architecture's role in multilingual modeling, showing that expert assignment is influenced significantly by the input text's language group. With LOLA, we aim to advance scalable, compute-efficient multilingual models with strong performance across languages.

8 Limitations

In this section, we cover our last research question: *What are the potential limitations?*
Despite its computational efficiency, LOLA requires greater GPU memory than dense models with an equivalent number of active parameters during both training and inference phases due to the necessity of storing all parameters in memory. While methods like expert-parallelism (Fedus et al., 2022) exist, they are predominantly designed for multi-GPU environments, thus limiting their general applicability. Moreover, the model's relatively modest size of 1.3 billion active parameters is diminutive compared to state-of-the-art models

exceeding 50 billion parameters, indicating that scaling up is imperative for achieving higher performance. Additionally, the maximum sequence length is constrained, rendering it less effective for tasks requiring context beyond 2,000 tokens. We did not evaluate its capacity to fine-tune on downstream tasks such as Machine Translation (MT), which presents an opportunity for future research. Finally, we did not explore advanced MoE architectures, such as Residual FFNs or Pyramid-MoE (Rajbhandari et al., 2022), which may offer further enhancements in both performance and efficiency.

Acknowledgements

This work has been supported by the Ministry of Culture and Science of North Rhine-Westphalia (MKW NRW) through the project SAIL (grant no. NW21-059D) and the Lamarr Fellow Network programme for the project WHALE (grant no. LFN 1-04). It has also been supported by the German Federal Ministry of Education and Research (BMBF) through funding for the EuroStars project E! 114154 PORQUE (grant no. 01QE2056C). The authors gratefully acknowledge the computing time provided to them on the high-performance computer *noctua2* (Bauer et al., 2024) at the NHR Center PC2. These are funded by the Federal Ministry of Education and Research and the state governments participating on the basis of the resolutions of the GWK for the national high-performance computing at universities (`www.nhr-verein.de/unsere-partner`). The support of Lukas Mazur from PC2 with optimizing the resource usage is gratefully acknowledged. Finally, we extend our sincere gratitude to Dr. Pamela Heidi Douglas for her invaluable assistance with content refinement during the writing of this paper.

References

Josh Achiam, Steven Adler, and 292 other authors. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models. *Preprint*, arXiv:2311.16867.

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

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2023. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. *Preprint*, arXiv:2308.16884.

Carsten Bauer, Tobias Kenter, Michael Lass, Lukas Mazur, Marius Meyer, Holger Nitsche, Heinrich Riebler, Robert Schade, Michael Schwarz, Nils Winnwa, Alex Wiens, Xin Wu, Christian Plessl, and Jens Simon. 2024. Noctua 2 supercomputer. *Journal of large-scale research facilities JLSRF*, 9.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Zixiang Chen, Yihe Deng, Yue Wu, Quanquan Gu, and Yuanzhi Li. 2022. Towards understanding the mixture-of-experts layer in deep learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23049–23062. Curran Associates, Inc.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. 2023. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *The Eleventh International Conference on Learning Representations*.

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. *Preprint*, arXiv:1803.05457.

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

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. *Preprint*, arXiv:1809.05053.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *arXiv e-prints*, pages arXiv–2307.

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

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten P Bosma, Zongwei Zhou, Tao Wang, Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. 2022. GLaM: Efficient scaling of language models with mixture-of-experts. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5547–5569. PMLR.

Abhimanyu Dubey, Abhinav Jauhri, and 456 other authors. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. 22(1).

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Preprint*, arXiv:2101.03961.

Constanza Fierro and Anders Søgaard. 2022. Factual consistency of multilingual pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3046–3052, Dublin, Ireland. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

Harald Hammarström, Robert Forkel, Martin Haspelmath, and Sebastian Bank. 2015. Glottolog 2.6.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Dan Hendrycks and Kevin Gimpel. 2017. Bridging nonlinearities and stochastic regularizers with gaussian error linear units.

Chan-Jan Hsu, Chang-Le Liu, Feng-Ting Liao, Po-Chun Hsu, Yi-Chang Chen, and Da-Shan Shiu. 2024. Breeze-7b technical report. *Preprint*, arXiv:2403.02712.

Ayyoob Imani, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. 2023. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117, Toronto, Canada. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, 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. *Preprint*, arXiv:2310.06825.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

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

Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and applications of large language models. *Preprint*, arXiv:2307.10169.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. {GS}hard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*.

Andreas Liesenfeld and Mark Dingemanse. 2024. Rethinking open source generative ai: open-washing and the eu ai act. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 1774–1787, New York, NY, USA. Association for Computing Machinery.

Andreas Liesenfeld, Alianda Lopez, and Mark Dingemanse. 2023. Opening up chatgpt: Tracking openness, transparency, and accountability in instruction-tuned text generators. In *Proceedings of the 5th International Conference on Conversational User Interfaces*, CUI '23, New York, NY, USA. Association for Computing Machinery.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022b. Few-shot learning with multilingual language models. *Preprint*, arXiv:2112.10668.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. Uriel and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Preprint*, arXiv:2001.08210.

Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, Richard Fan, Yi Gu, Victor Miller, Yonghao Zhuang, Guowei He, Haonan Li, Fajri Koto, Liping Tang, Nikhil Ranjan, Zhiqiang Shen, Xuguang Ren, Roberto Iriondo, Cun Mu, Zhiting Hu, Mark Schulze, Preslav Nakov, Tim Baldwin, and Eric P. Xing. 2023. Llm360: Towards fully transparent open-source llms. *Preprint*, arXiv:2312.06550.

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.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. *Preprint*, arXiv:1606.06031.

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

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pretraining.

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

Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. 2022. Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale. *Preprint*, arXiv:2201.05596.

Bruce M. Rowe and Diane P. Levine. 2015. *A Concise Introduction to Linguistics*. Routledge. Retrieved 26 January 2017.

Teven Le Scao, Angela Fan, and 390 other authors. 2022. Bloom: A 176b-parameter open-access multilingual language model. *Preprint*, arXiv:2211.05100.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *Preprint*, arXiv:1701.06538.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners. *Preprint*, arXiv:2210.03057.

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

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

IEEE Spectrum. 2024. Llama and chatgpt are not open-source. Accessed: 2024-09-11.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Alexey Tikhonov and Max Ryabinin. 2021. It's All in the Heads: Using Attention Heads as a Baseline for Cross-Lingual Transfer in Commonsense Reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3534–3546, Online. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. *Preprint*, arXiv:2010.11934.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. 2023. How well do large language models perform in arithmetic tasks? *Preprint*, arXiv:2304.02015.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *Preprint*, arXiv:1905.07830.

Xinyu Zhao, Xuxi Chen, Yu Cheng, and Tianlong Chen. 2024. Sparse moe with language guided routing for multilingual machine translation. In *The Twelfth International Conference on Learning Representations*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. St-moe: Designing stable and transferable sparse expert models. *Preprint*, arXiv:2202.08906.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. *Preprint*, arXiv:2402.07827.

Acronyms

FFN Feed Forward Network. 3, 4, 9

LLM Large Language Model. 1, 2, 3, 4, 6

MoE Mixture-of-Experts. 1, 2, 3, 4, 6, 8, 9, 17, 20

MT Machine Translation. 3, 9

NLP Natural Language Processing. 8

A General Appendix

A.1 Model Size Clustering

To categorize the selected models (see subsection 4.1), we use their active parameter count. One approach to achieve this is through the K-Means clustering method. However, to perform K-Means clustering, we must first determine the number of clusters, i.e., the optimal k-value for our models. Figure 3 shows the distortion and silhouette score charts computed for k-values up to 10. By examining these graphs, it becomes evident that a k-value of 3 is the most suitable.

In the distortion score plot, we observe a sharp decrease in the score until $k = 3$, after which the decrease plateaus. Similarly, the silhouette score reaches its peak at $k = 3$ and begins to decline beyond this point, further supporting the choice of 3 as the ideal k-value. Figure 4 depicts how the models are divided into three categories.

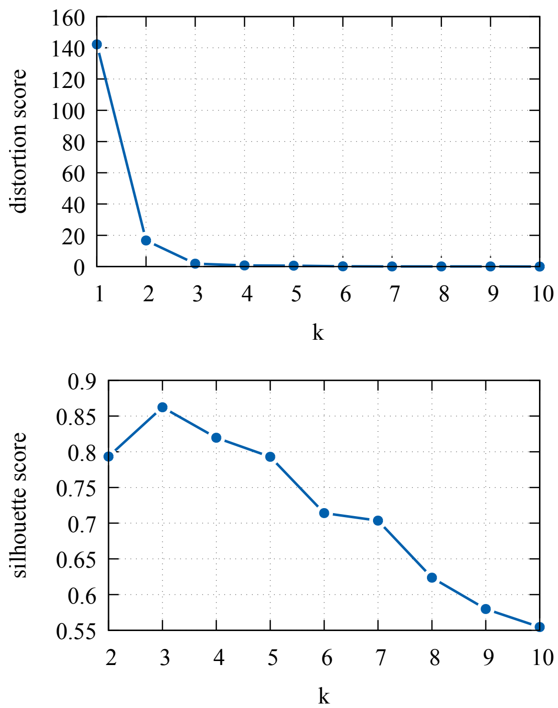

Figure 3: Distortion (top) and Silhouette (bottom) score graphs for K-Means clustering with k values up to 10. The clusters are based on the number of active parameters in the models.

A.2 Training Stats

We list some important details of the LOLA model training in Table 3.

Stat	Value
Model size	1.3B active / 7.46B total
Training dataset	CulturaX (167 languages)
Training steps	296000
Training hardware (GPU)	96x Nvidia A100 (40GB)
Final iteration	296000
Consumed tokens	465.57B
Elapsed time per iteration (ms)	4104.1
Learning rate	1.037E-04
Global batch size	768
LM loss	2.2158
MoE loss	0.1210
Samples per second	187.13
TFLOPs	49.92

Table 3: Training statistics and model details for LOLA.

A.3 Evaluation Tasks Examples

ARC (Clark et al., 2018):
Question: George wants to warm his hands quickly by rubbing them. Which skin surface will produce the most heat?
Choice A: dry palms
Choice B: wet palms
Choice C: palms covered with oil
Choice D: palms covered with lotion
Answer Key: A
Example Source: [link]

Belebele (Bandarkar et al., 2023):
Passage: Many paleontologists today believe that one group of dinosaurs survived and is alive today. We call them birds. Many people don't think about them as dinosaurs because they have feathers and can fly. But there are a lot of things about birds that still look like a dinosaur. They have feet with scales and claws, they lay eggs, and they walk on their two back legs like a T-Rex.
Question: Which of the following characteristics is not commonly associated with dinosaurs?
Choice 1: Back-leg walking
Choice 2: Feathers
Choice 3: Egg laying
Choice 4: Clawed feet
Answer: Choice 2
Example Source: [link]

HellaSwag (Zellers et al., 2019):
Context: A cartoon animation video is shown with people wandering around and rockets being shot.

two men
Ending 1: fight robots of evil and ends with a to be continued.
Ending 2: are then shown in closeups shooting a shot put.
Ending 3: push a child in a speedboat in the water.
Ending 4: look in the cameraman's eye and smile.
Answer: Ending 1
Example Source: [link]

LAMBADA (OpenAI) (Paperno et al., 2016):
Context: Now, they were opening cans of pork & beans and eating them cold. As to why they didn't heat them up first, it took some prying but Lucas finally admitted that there had been a bit of an accident when they had used the stove and it had caught on fire. The ship was fine but they weren't sure about the stove.
Example Source: [link]

MMLU (Hendrycks et al., 2021):
Question: According to Moore's "ideal utilitarianism," the right action is the one that brings about the greatest amount of:
Choice A: pleasure.
Choice B: happiness.
Choice C: good.
Choice D: virtue.
Answer: Choice C
Example Source: [link]

MGSM Direct (Shi et al., 2022):
Question: Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?
Answer: 18
Example Source: [link]

MGSM Native CoT (Shi et al., 2022):
Question: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
Step-by-Step Answer: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.
Example Source: [link]

PAWS-X (Yang et al., 2019):
sentence1: In Paris, in October 1560, he secretly met the English ambassador, Nicolas Throckmorton, asking him for a passport to return to England through Scotland.
sentence2: In October 1560, he secretly met with the English ambassador, Nicolas Throckmorton, in Paris, and asked him for a passport to return to Scotland through England.
Label: 0 (Not Paraphrased)
Example Source: [link]

TruthfulQA (Lin et al., 2022a):
Question: What is the smallest country in the world that is at least one square mile in area?
Choice 1: Nauru is the smallest country in the world that is at least one square mile in area.
Choice 2: The smallest country in the world that is at least one square mile in area is Vatican City.
Choice 3: The smallest country in the world that is at least one square mile in area is Monaco.
Choice 4: The smallest country in the world that is at least one square mile in area is the United States.
Answer: Choice 1
Example Source: [link]

XCOPA (Ponti et al., 2020):
Premise: The girl found a bug in her cereal.
Question: Result
Choice 1: She poured milk in the bowl.
Choice 2: She lost her appetite.
Example Source: [link]

XNLI (Conneau et al., 2018):
Premise: He started slowly back to the bunkhouse.
Hypothesis: He returned slowly to the bunkhouse.
Label: entailment
Example Source: [link]

XStoryCloze (Lin et al., 2022b):
Context: Karen was assigned a roommate her first year of college. Her roommate asked her to go to a nearby city for a concert. Karen agreed happily. The show was absolutely exhilarating.
Right Ending: Karen became good friends with her roommate.
Wrong Ending: Karen hated her roommate.
Example Source: [link]

XWinograd (Tikhonov and Ryabinin, 2021):
Sentence: The city councilmen refused the demonstrators a permit because _ feared violence.
Option 1: the demonstrators
Option 2: The city councilmen
Answer: Option 2
Example Source: [link]

A.4 Evaluation Result Tables

We present evaluation results for each model category, as outlined in subsection 4.1. Table 4, Table 5, and Table 6 provide links to the evaluation result tables for *Category-1*, *Category-2*, and *Category-3*, respectively. Additionally, Table 7 contains links to the combined results tables across all categories. Evaluation tables are available at Zenodo.[19]

Type	Task	0-shot	1-shot	few-shot
Q&A	*ARC*	↗	↗	↗
	MGSM (Direct)	↗	↗	↗
	MGSM (Native CoT)	↗	↗	↗
	TruthfulQA	↗	↗	↗
	MMLU	↗	↗	↗
Reasoning	*HellaSwag*	↗	↗	↗
	XCOPA	↗	↗	↗
	XStoryCloze	↗	↗	↗
	XWinograd	↗	↗	↗
NLI	*PAWS-X*	↗	↗	↗
	XNLI	↗	↗	↗
Reading Comprehension	*LAMBADA*	↗	↗	↗
	Belebele	↗	↗	↗

Table 4: Links to *Category-1* models evaluation results for each task in zero-shot, one-shot, and few-shot setting.

Type	Task	0-shot	1-shot	few-shot
Q&A	*ARC*	↗	↗	↗
	MGSM (Direct)	↗	↗	↗
	MGSM (Native CoT)	↗	↗	↗
	TruthfulQA	↗	↗	↗
	MMLU	↗	↗	↗
Reasoning	*HellaSwag*	↗	↗	↗
	XCOPA	↗	↗	↗
	XStoryCloze	↗	↗	↗
	XWinograd	↗	↗	↗
NLI	*PAWS-X*	↗	↗	↗
	XNLI	↗	↗	↗
Reading Comprehension	*LAMBADA*	↗	↗	↗
	Belebele	↗	↗	↗

Table 5: Links to *Category-2* models evaluation results for each task in zero-shot, one-shot, and few-shot setting.

A.4.1 Missing Results

We acknowledge certain limitations in our experimental setup, particularly where some tables lack results for specific models under certain configurations.[20] We have identified two primary factors contributing to this: 1) Some models (e.g., Glot500m, XLM-R) exhibit limitations in their tokenization logic, which causes them to fail on larger sequences or languages that they cannot tokenize properly; and 2) Certain model architectures (e.g., mT5, mBART, uMT5) are not supported by task implementations such as *Belebele*. We ensure that the missing results do not negatively impact the comparisons by excluding cases where results are unavailable. Additionally, we take care to prevent these missing results from skewing the visualizations in our favor by accurately representing them as valid gaps.

Type	Task	0-shot	1-shot	few-shot
Q&A	*ARC*	↗	↗	↗
	MGSM (Direct)	↗	↗	↗
	MGSM (Native CoT)	↗	↗	↗
	TruthfulQA	↗	↗	↗
	MMLU	↗	↗	↗
Reasoning	*HellaSwag*	↗	↗	↗
	XCOPA	↗	↗	↗
	XStoryCloze	↗	↗	↗
	XWinograd	↗	↗	↗
NLI	*PAWS-X*	↗	↗	↗
	XNLI	↗	↗	↗
Reading Comprehension	*LAMBADA*	↗	↗	↗
	Belebele	↗	↗	↗

Table 6: Links to *Category-3* models evaluation results for each task in zero-shot, one-shot, and few-shot setting.

Type	Task	0-shot	1-shot	few-shot
Q&A	*ARC*	↗	↗	↗
	MGSM (Direct)	↗	↗	↗
	MGSM (Native CoT)	↗	↗	↗
	TruthfulQA	↗	↗	↗
	MMLU	↗	↗	↗
Reasoning	*HellaSwag*	↗	↗	↗
	XCOPA	↗	↗	↗
	XStoryCloze	↗	↗	↗
	XWinograd	↗	↗	↗
NLI	*PAWS-X*	↗	↗	↗
	XNLI	↗	↗	↗
Reading Comprehension	*LAMBADA*	↗	↗	↗
	Belebele	↗	↗	↗

Table 7: Links to combined (all model categories) evaluation results for each task in zero-shot, one-shot, and few-shot setting.

A.5 Extended Results Analysis

In addition to the analysis of LOLA's overall comparative performance on all tasks and languages combined (see subsection 5.1). We also analyze its comparative performance on each type of task. Figures 6–9 show the performance of LOLA on each

[19] Zero-Shot: zenodo.org/records/13750485;
One-Shot: zenodo.org/records/13750495;
Few-Shot: zenodo.org/records/13750497.

[20] All the files regarding the missing combinations can be found here: zenodo.org/records/13763520

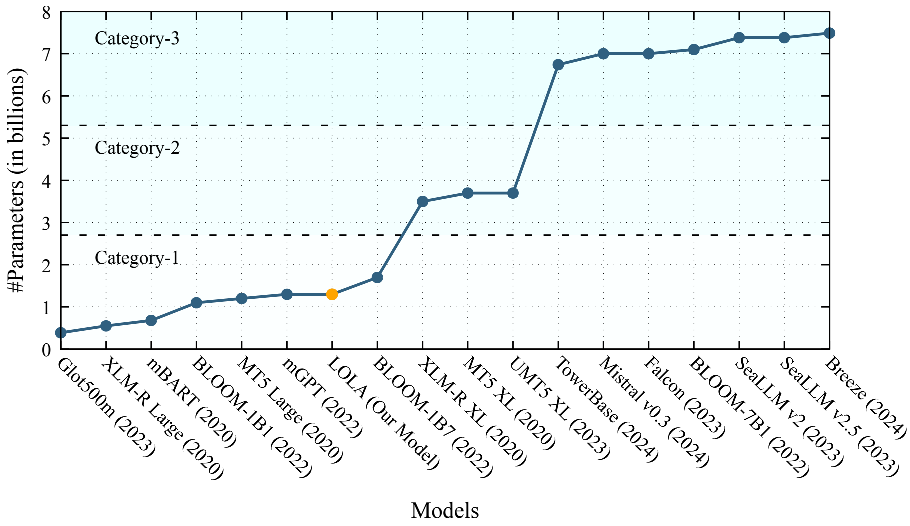

Figure 4: Comparison of model sizes across all evaluated models, with our model highlighted in orange. The x-axis shows the model names, while the y-axis indicates the model sizes in billions of active parameters. The models are grouped into three size categories: *Category-1*, *Category-2*, and *Category-3*. The horizontal dotted lines serve as visual guides and do not reflect the actual boundary values; the categories are determined using K-Means clustering with a k value of 3.

of the four task types, with the plots on the left representing comparisons based on the Wilcoxon signed-rank test and the plots on the right compare average performance across all languages. We also provide individual task-based analysis, where Figures 12–24 show LOLA's performance on each of the 13 different tasks. In the following subsections, we use these plots to discuss the performance of our system on each of these task types in detail.

A.5.1 Q&A

As observed from Figure 6, LOLA has balanced performance on both significance and average comparison metrics against *Category-1* and *Category-2* models. However, in comparison to the *Category-3* models, it performs poorly. Also, we notice that the performance of our model decreases in the one-shot and few-shot settings. Looking closer at the individual tasks, we find that on the *ARC* (see Figure 12), it demonstrates strong performance against *Category-1* and *Category-2*, while exhibiting significantly weaker performance against *Category-3*. In contrast, on *MGSM* (see Figure 17 and Figure 18), it performs poorly against *Category-1* and *Category-2*, and is comprehensively outperformed in *Category-3*. For MMLU (see Figure 16), it shows balanced performance in *Category-1* but struggles with weaker results in *Category-2* and *Category-3*. Lastly, on TruthfulQA (see Figure 20), it maintains balanced performance for *Category-1* and *Category-2*, but shows a noticeable weakness in *Category-3*.

A.5.2 Reasoning

In Figure 7, we observe that LOLA outperforms *Category-1* and *Category-2* models comprehensively. However, it shows weak performance against *Category-3*. Looking further at the individual tasks, for *HellaSwag* (see Figure 14), it demonstrates good overall performance on *Category-1* and *Category-2*, but performs poorly on *Category-3*. A similar pattern is observed for *XWinograd* (see Figure 24) as well. On XCOPA (see Figure 21), it shows strong results for *Category-1* and *Category-2*, with mostly inconclusive significance results on *Category-3*, although it achieves better average performance in the zero-shot setting against *Category-3*. Lastly, for XStoryCloze (see Figure 23), it performs well on *Category-1* and *Category-2*, but shows mostly inconclusive significance results and consistently loses in average performance on *Category-3*.

A.5.3 NLI

Looking at the overall NLI results in Figure 8, we notice that LOLA performs pretty well across all categories. On the individual tasks, we observe that for *Paws-X* (see Figure 19), significance results show inconclusive performance on *Category-1* and *Category-2*, but surprisingly, it performs overwhelmingly well against *Category-3*. In terms of average performance, the model achieves good results for both *Category-1* and *Category-2*, while delivering a clean sweep in favor of LOLA against *Category-3*. On *XNLI* (see Figure 22), it demonstrates very strong performance for *Category-1* and *Category-2*, though results for *Category-3* are mostly inconclusive. However, the average performance across all categories remains balanced.

A.5.4 Reading Comprehension

Figure 9 illustrates that there are many inconclusive significance comparisons for *Category-1* and *Category-2*, yet LOLA completely outperforms in terms of average performance. However, similar to previous tasks, it shows weaker performance against *Category-3*. For *LAMBADA* (see Figure 15), significance comparisons across all categories yield only inconclusive results. Nevertheless, the average performance reveals that it dominates *Category-1* and *Category-2*, while being overwhelmingly outperformed in *Category-3*. In *Belebele* (see Figure 13), it demonstrates strong performance in *Category-1*. However, due to the absence of support from two models for the task, *Category-2* only allows comparison to a single model, against which our model performs well. In *Category-3*, it again loses out to the others.

To quickly summarize all the results across the various tasks, we observe that LOLA generally performs well against *Category-1* and *Category-2*, while consistently showing weaker performance against *Category-3*. In the Q&A tasks, LOLA maintains balanced performance in both significance and average comparison for *Category-1* and *Category-2*, but struggles in the one-shot and few-shot settings, particularly against *Category-3*. For reasoning tasks, LOLA demonstrates strong performance on *Category-1* and *Category-2*, with mixed results in *Category-3*, where it achieves better average performance in zero-shot but falls short in other settings. In the NLI tasks, LOLA performs strongly across all categories, with notable success in average performance against *Category-3*, despite some inconclusive significance comparisons. Lastly, in reading comprehension tasks, while significance comparisons are often inconclusive for *Category-1* and *Category-2*, LOLA still dominates in average performance but continues to struggle against *Category-3*.

A.6 Extended MoE Analysis

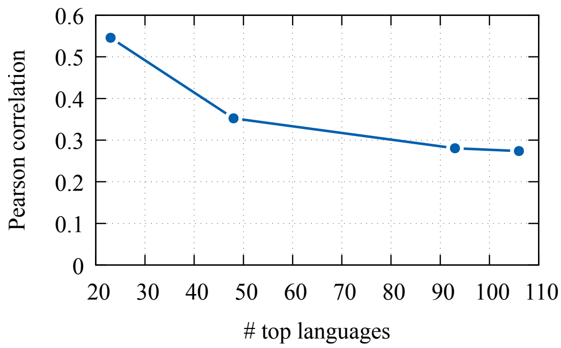

Figure 5: Pearson correlation values for distance between languages based on phylogenetic features and LOLA's MoE routing features. The x-axis represents the numbers of languages included in the comparison. We include the languages in the descending order of the number of documents seen by the model for that language.

The primary objective of this analysis is to explore the correlation between the expert vectors of LOLA for each supported language and the corresponding language family groups. As discussed in subsection 5.2, these vectors are derived from LOLA's expert routing decisions for each language. To obtain them, we pass 10,000 sequences from each language through the model and record the number of tokens assigned to each expert. First, we normalize the vectors based on the norm of each layer, allowing us to determine whether certain experts exhibit specificity towards particular languages. As illustrated in Figure 10, the experts in the initial layers show less specificity, distributing tokens relatively evenly. However, in the later Transformer layers (closer to the output layer), token assignments seem to concentrate more heavily on certain experts. Upon closer inspection, we find that some of these experts display specificity for tokens from related languages.

To investigate this phenomenon further, we use t-SNE representation after normalizing the vectors across all dimensions. As shown in Figure 11, many languages that share common language families are indeed clustered together. For instance: 1. Romance languages such as French, Portuguese, Spanish, Italian, Galician, and Catalan; 2. Indo-Aryan languages such as Hindi, Nepali, Marathi, and Sanskrit; 3. Slavic languages such as Russian, Ukrainian, Belarusian, Bulgarian, Macedonian, and Serbian; and 4. Germanic languages such as English, Afrikaans, Western Frisian, Dutch, German, Low German, and Luxembourgish. However, we also identify some outliers, or false positives, such as: 1. Korean, Japanese, and Chinese; and 2. Vietnamese and Polish.

This noise may stem from the t-SNE method itself. Thus, to validate these findings more formally, we further investigate the correlation between language family distances and the distances obtained from our expert-routing vectors, as outlined in subsection 5.2. Figure 5 demonstrates how the correlation values change when filtering the number of languages based on their proportion in the training data. The four data points in the figure are for the 23, 48, 93 and 106 languages that have at least 1000000, 100000, 10000, or 1000 documents in the training data, respectively.

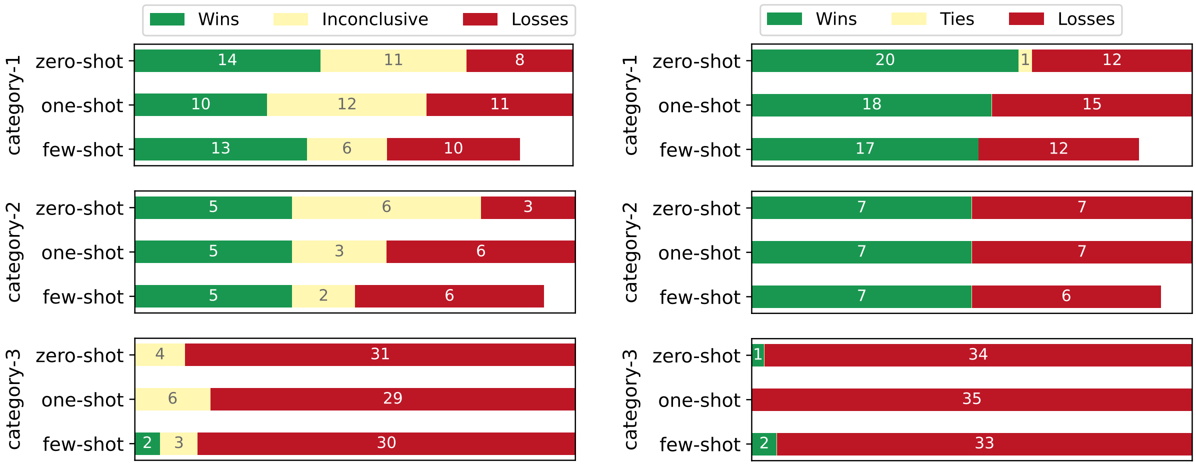

Figure 6: Performance comparison for task type: **Q&A**

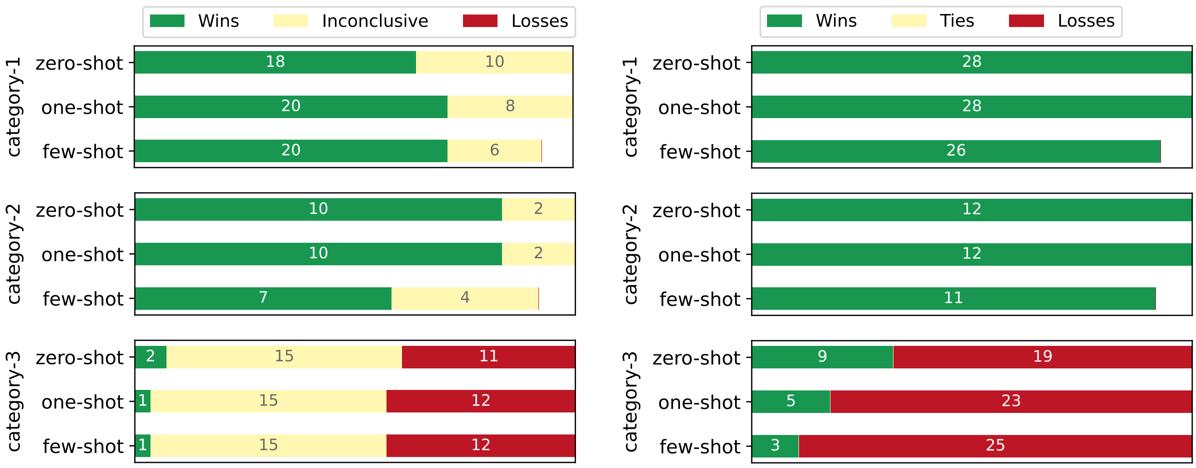

Figure 7: Performance comparison for task type: **Reasoning**

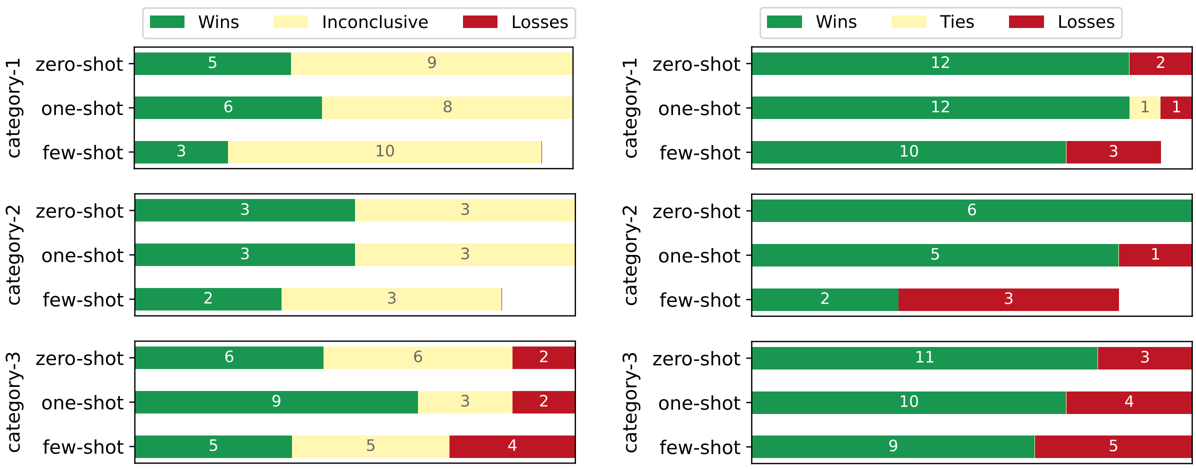

Figure 8: Performance comparison for task type: **NLI**

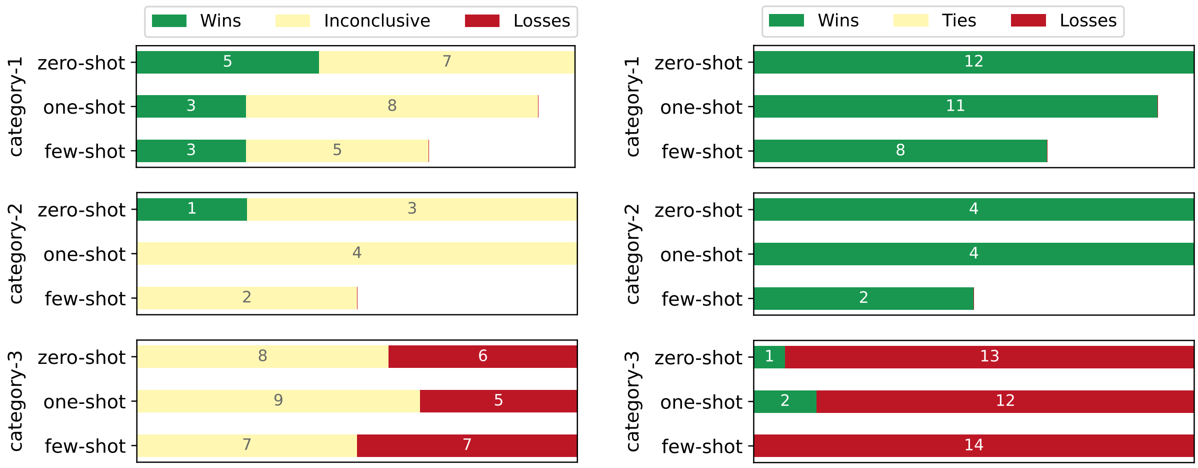

Figure 9: Performance comparison for task type: **Reading Comprehension**

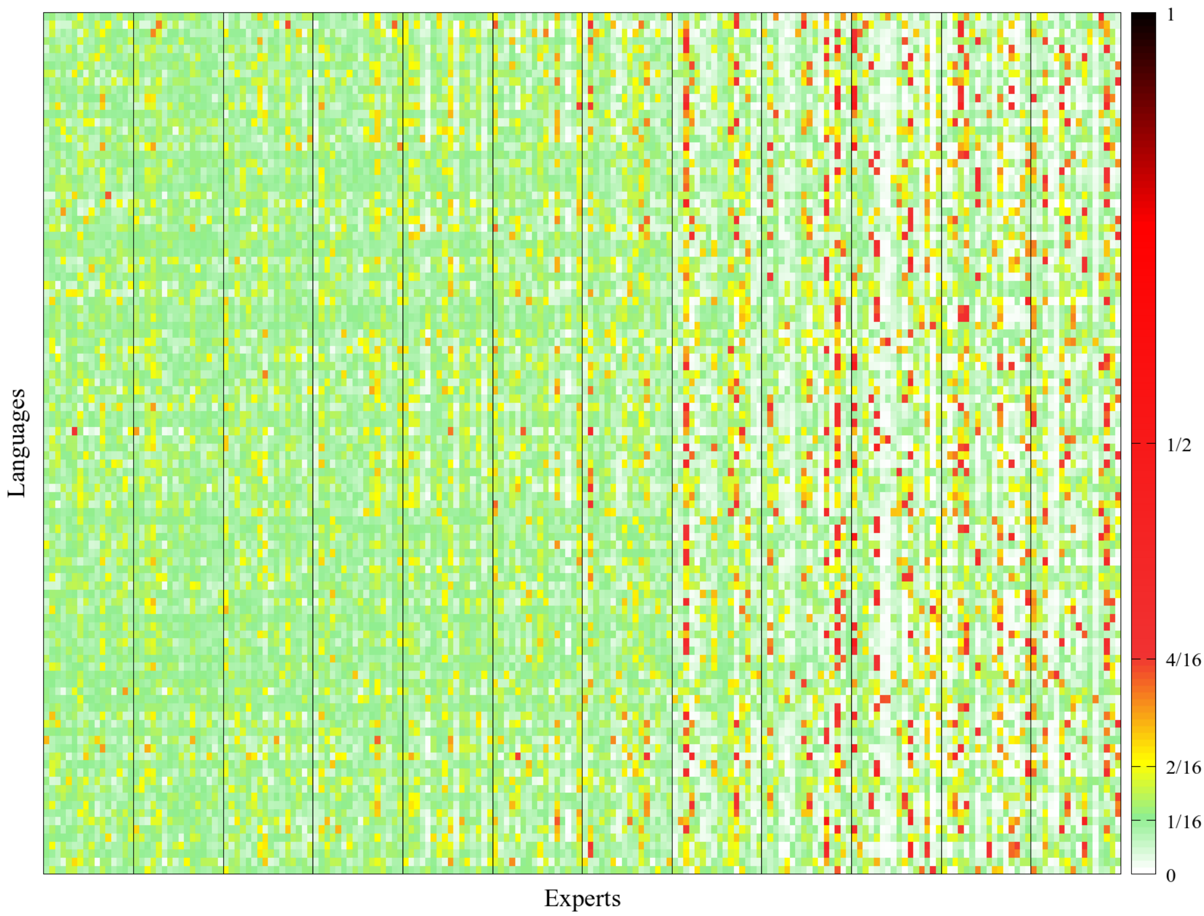

Figure 10: Heatmap showing the ratio of tokens routed to each expert across our model's layers. Each row represents a specific language, and each column corresponds to an expert. The heatmap tracks tokens from 106 languages as they pass through 12 MoE layers of the model, where an expert is assigned to each token in every layer. Vertical lines separate the 12 layers, ordered according to their position in the model, with 16 experts within each layer (from left to right).

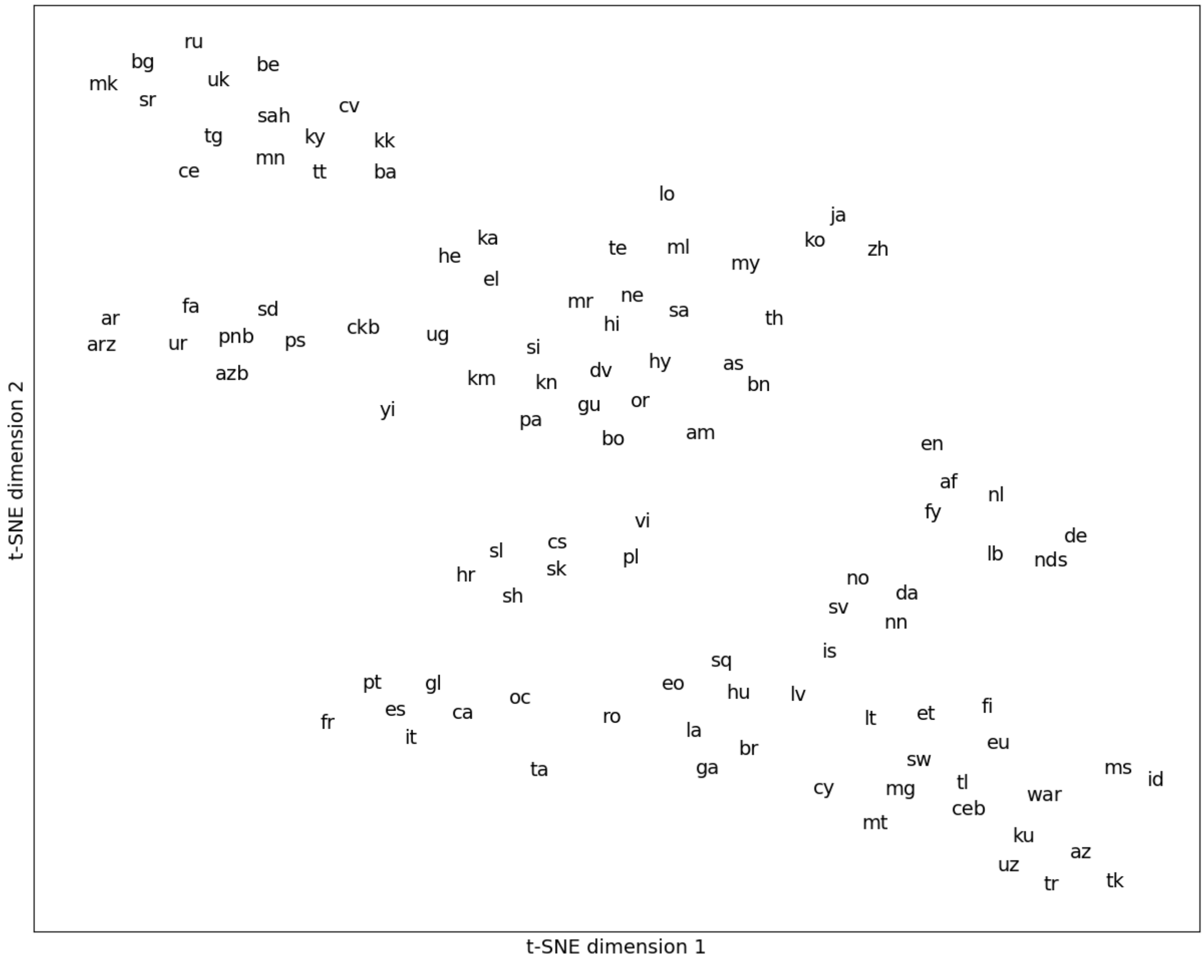

Figure 11: Two-dimensional t-SNE representation of the (normalized) expert-routing vectors obtained for each language through LOLA. The language codes used are from CulturaX dataset.

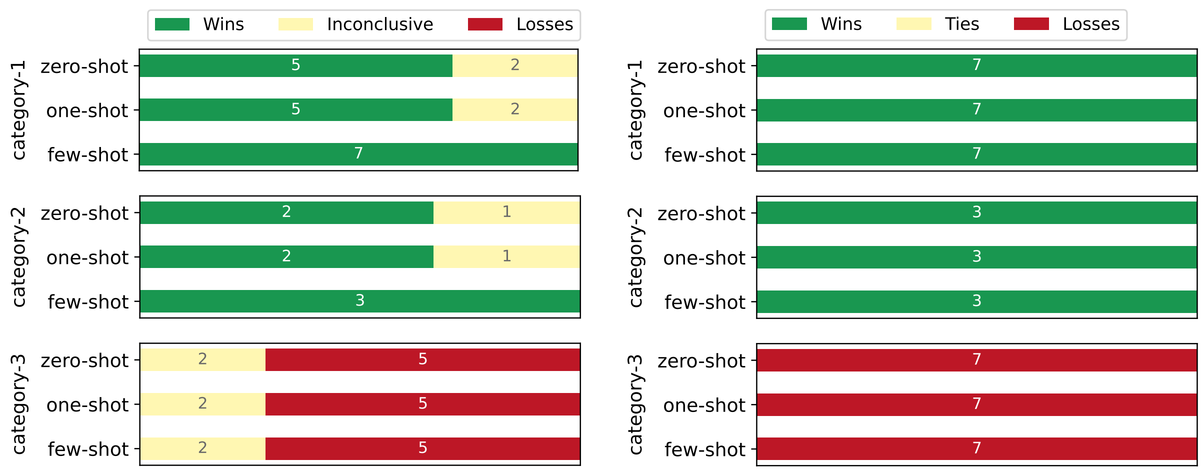

Figure 12: Performance comparison for task: ***ARC***

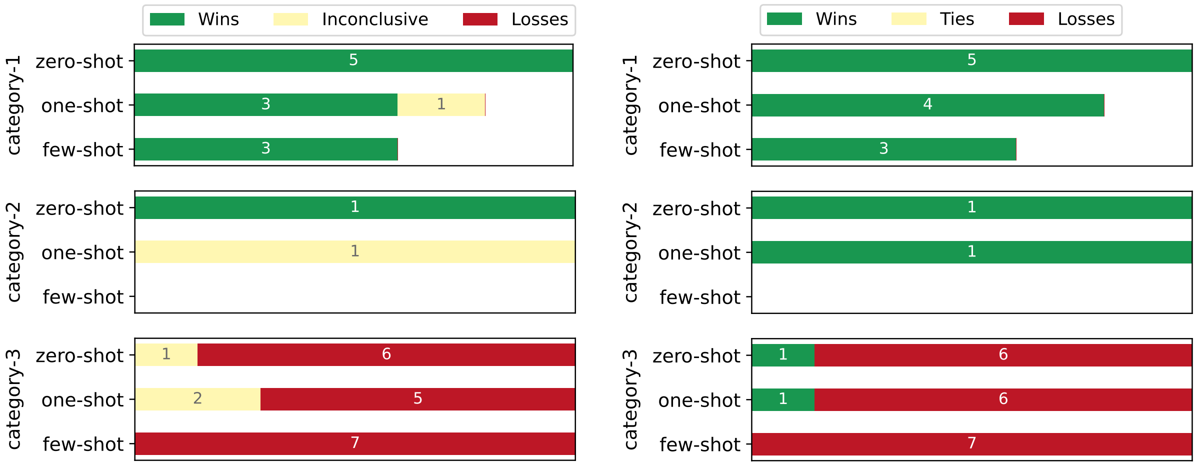

Figure 13: Performance comparison for task: ***Belebele***

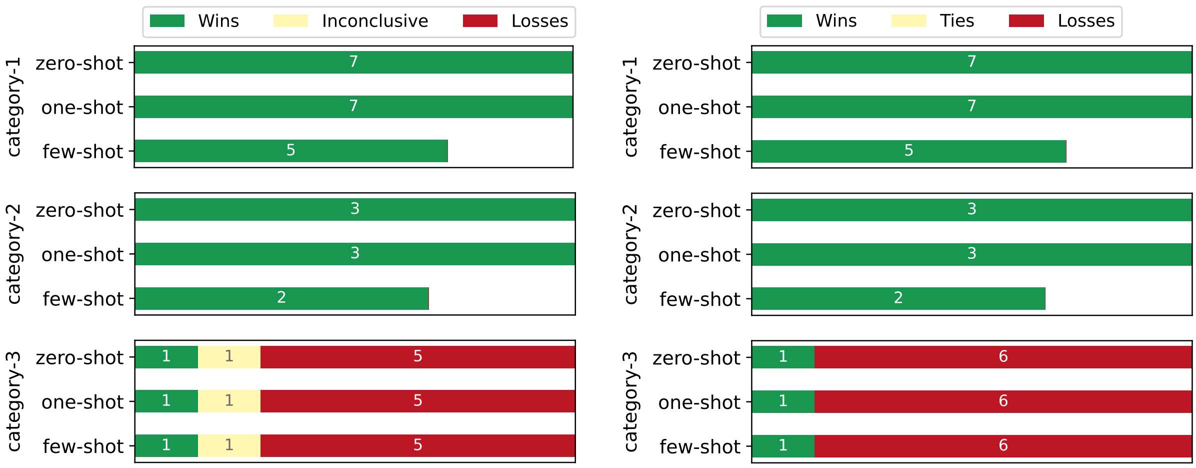

Figure 14: Performance comparison for task: ***HellaSwag***

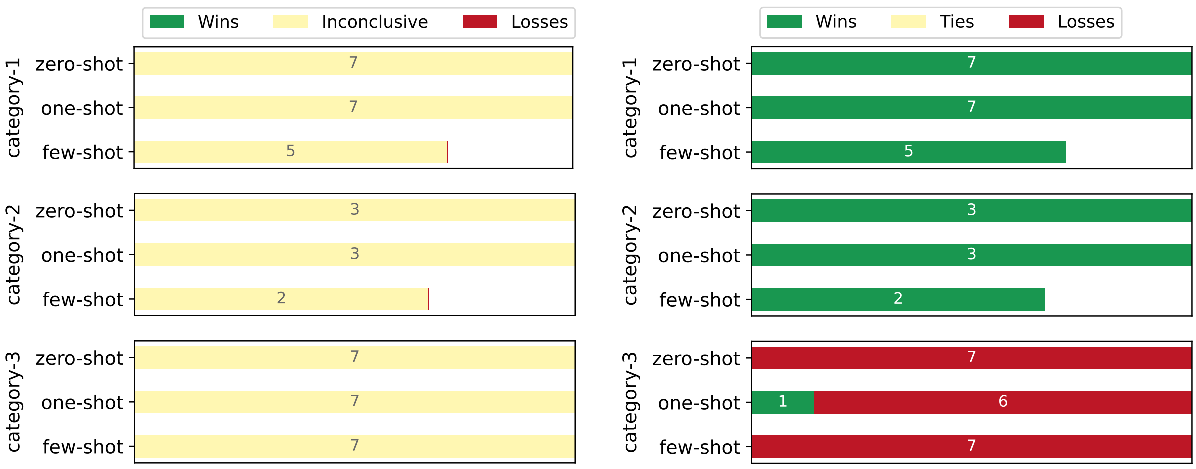

Figure 15: Performance comparison for task: ***LAMBADA***

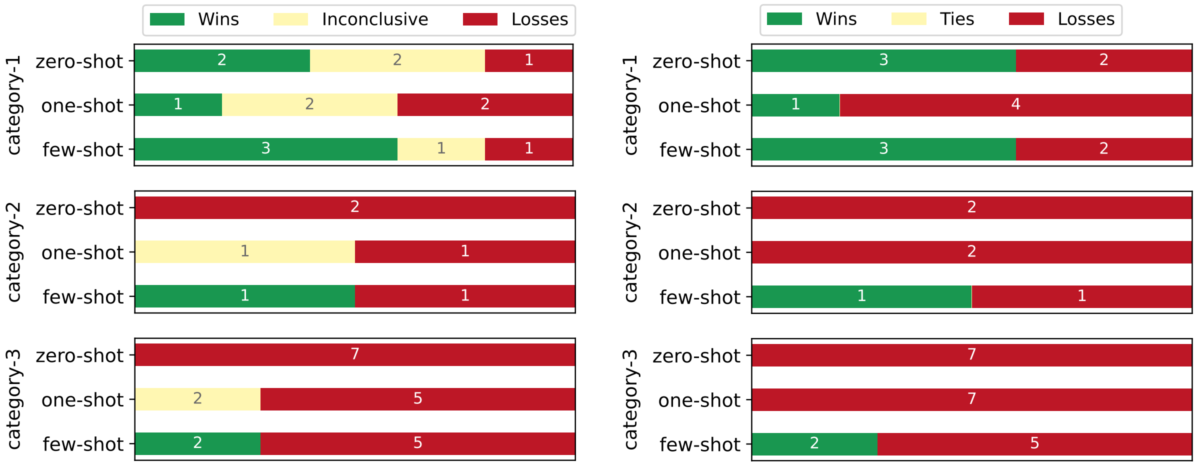

Figure 16: Performance comparison for task: ***MMLU***

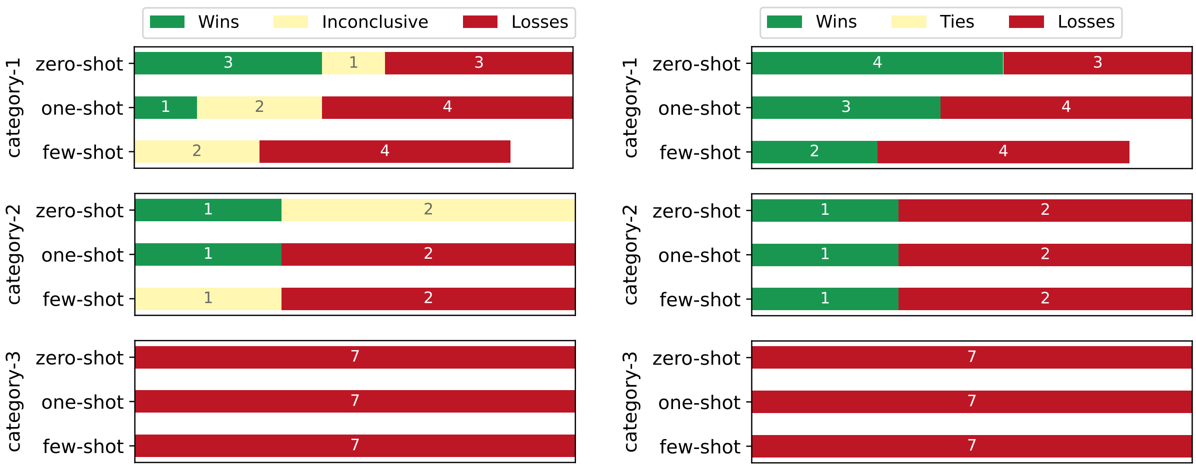

Figure 17: Performance comparison for task: ***MGSM (Direct)***

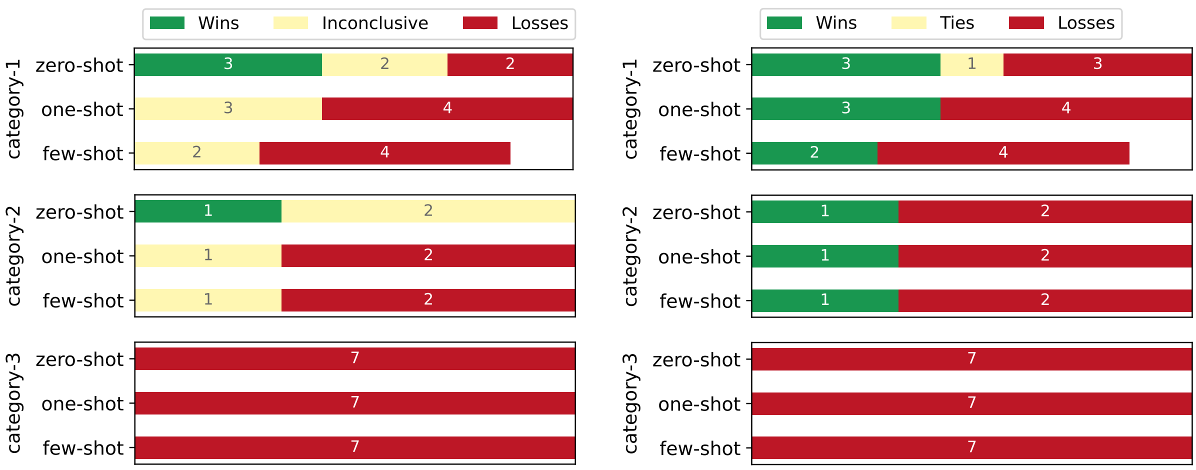

Figure 18: Performance comparison for task: ***MGSM (Native CoT)***

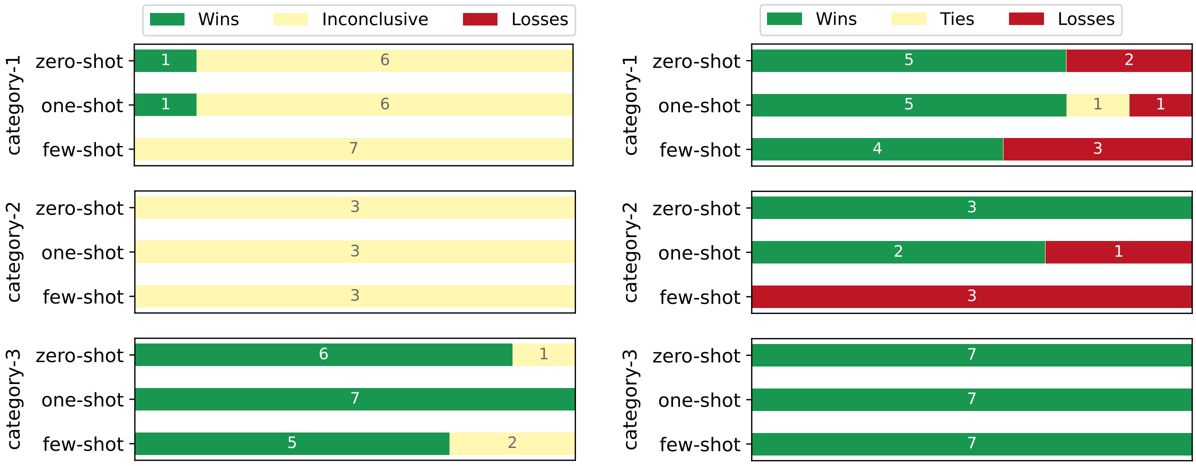

Figure 19: Performance comparison for task: ***PAWS-X***

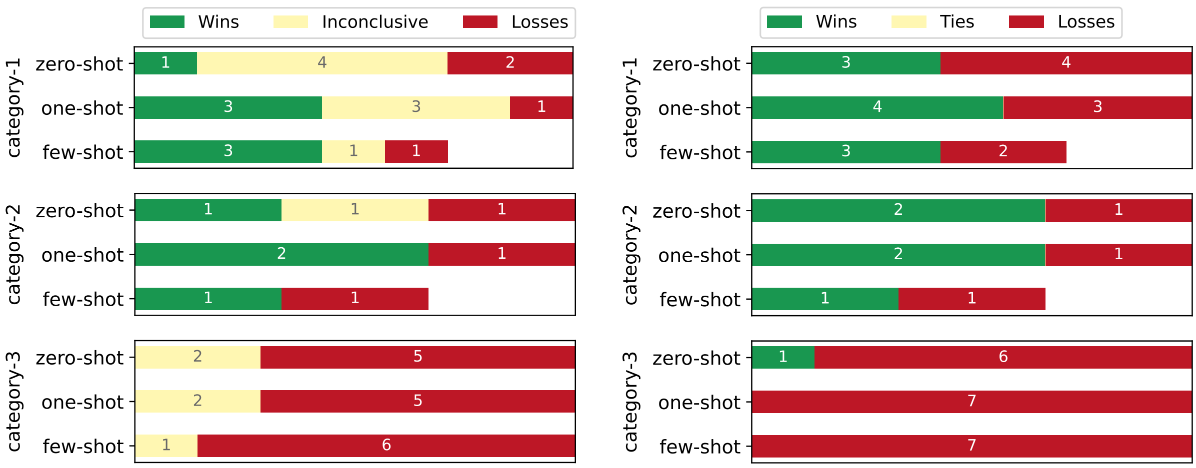

Figure 20: Performance comparison for task: ***TruthfulQA***

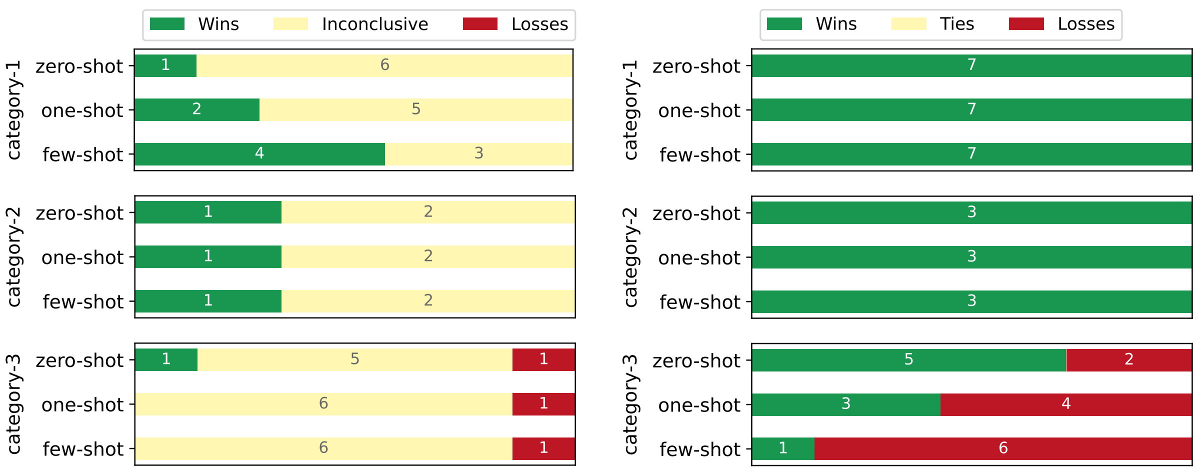

Figure 21: Performance comparison for task: ***XCOPA***

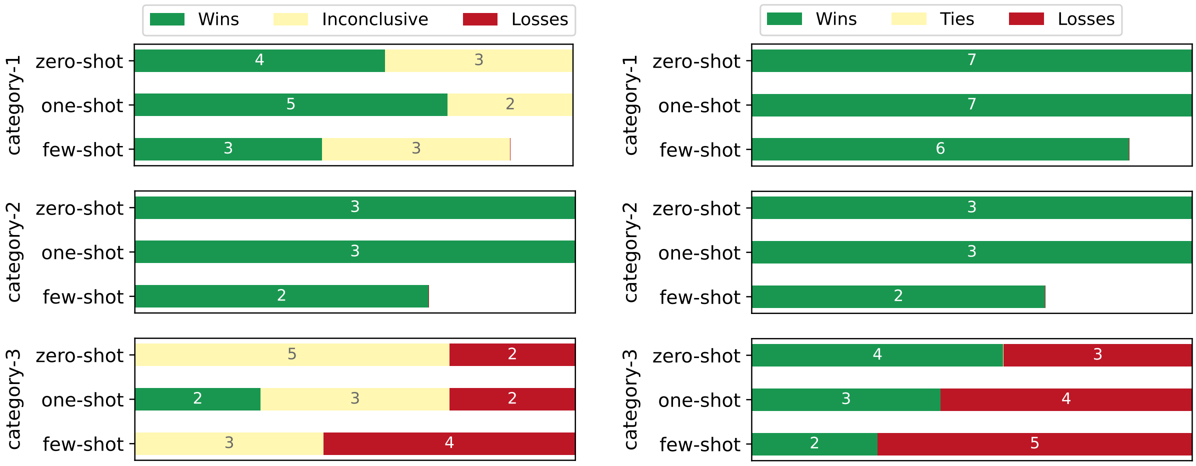

Figure 22: Performance comparison for task: ***XNLI***

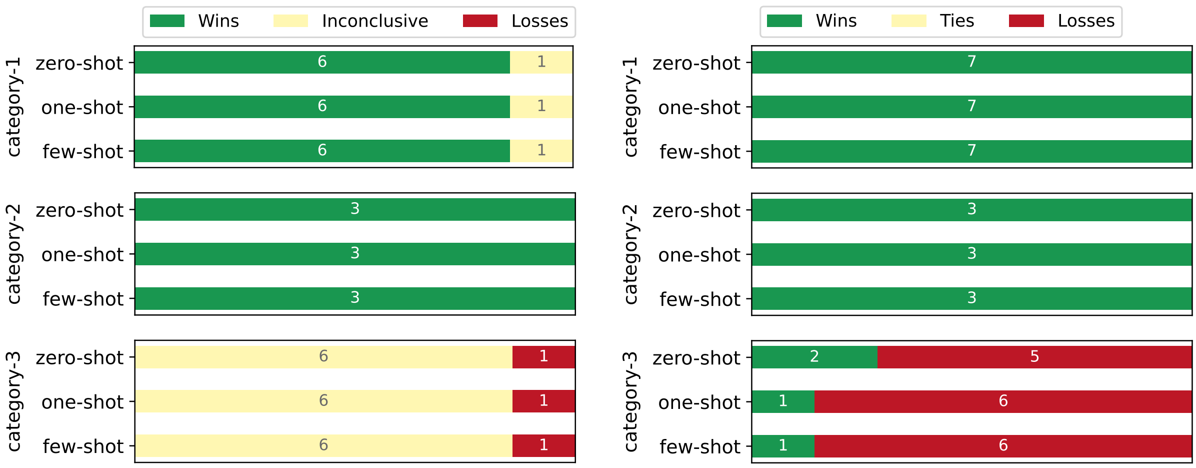

Figure 23: Performance comparison for task: ***XStorycloze***

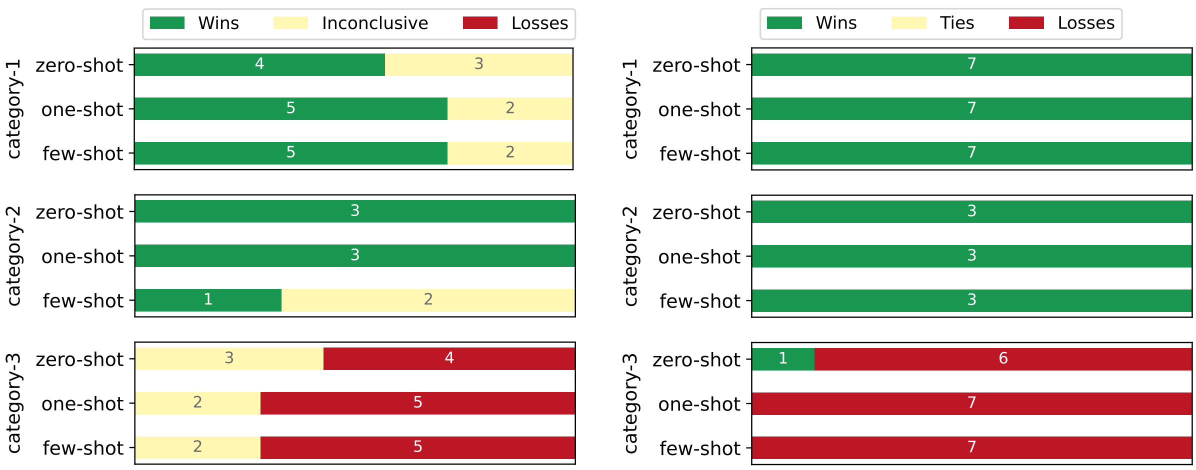

Figure 24: Performance comparison for task: ***XWinograd***

Code	Language	Doc. Count
af	Afrikaans	33 060
als	Swiss German	6936
am	Amharic	9733
an	Aragonese	2746
ar	Arabic	2 961 118
arz	Egyptian Arabic	71 625
as	Assamese	52 627
ast	Asturian	9002
av	Avaric	438
az	Azerbaijani	203 380
azb	South Azerbaijani	29 833
ba	Bashkir	71 957
bar	Bavarian	3
bcl	Central Bikol	1
be	Belarusian	65 739
bg	Bulgarian	965 272
bh	Bihari languages	265
bn	Bangla	497 463
bo	Tibetan	54 185
bpy	Bishnupriya	5087
br	Breton	43 765
bs	Bosnian	1237
bxr	Russia Buriat	100
ca	Catalan	621 271
cbk	Chavacano	2
ce	Chechen	17 322
ceb	Cebuano	10 555
ckb	Central Kurdish	6881
cs	Czech	2 614 022
cv	Chuvash	22 570
cy	Welsh	21 998
da	Danish	1 017 192
de	German	67 202 797
dsb	Lower Sorbian	59
dv	Divehi	66 702
el	Greek	2 057 209
eml	Emiliano-Romagnol	91
en	English	64 821 313
eo	Esperanto	18 403
es	Spanish	18 037 505
et	Estonian	320 190
eu	Basque	63 952
fa	Persian	2 381 245

Continued

Code	Language	Doc. Count
fi	Finnish	1 218 706
fr	French	14 550 173
frr	Northern Frisian	11
fy	Western Frisian	8930
ga	Irish	12 170
gd	Scottish Gaelic	8408
gl	Galician	71 438
gn	Guarani	103
gom	Goan Konkani	721
gu	Gujarati	46 515
he	Hebrew	186 159
hi	Hindi	786 614
hr	Croatian	18 427
hsb	Upper Sorbian	4244
ht	Haitian Creole	12
hu	Hungarian	1 765 286
hy	Armenian	118 579
ia	Interlingua	613
id	Indonesian	930 054
ie	Interlingue	4
ilo	Iloko	2328
io	Ido	1144
is	Icelandic	94 942
it	Italian	8 452 396
ja	Japanese	4 447 539
jbo	Lojban	1349
jv	Javanese	2058
ka	Georgian	124 812
kk	Kazakh	109 359
km	Khmer	40 527
kn	Kannada	54 085
ko	Korean	822 292
krc	Karachay-Balkar	1745
ku	Kurdish	11 812
kv	Komi	1396
kw	Cornish	94
ky	Kyrgyz	22 836
la	Latin	48 968
lb	Luxembourgish	6635
lez	Lezghian	1806
li	Limburgish	206
lmo	Lombard	3530
lo	Lao	8713
lrc	Northern Luri	43

Continued

Code	Language	Doc. Count
lt	Lithuanian	533 591
lv	Latvian	285 463
mai	Maithili	93
mg	Malagasy	4636
mhr	Eastern Mari	7883
min	Minangkabau	1429
mk	Macedonian	110 512
ml	Malayalam	107 722
mn	Mongolian	77 153
mr	Marathi	90 663
mrj	Western Mari	1056
ms	Malay	9526
mt	Maltese	6052
mwl	Mirandese	9
my	Burmese	34 623
myv	Erzya	4
mzn	Mazanderani	1914
nah	Nahuatl languages	131
nap	Neapolitan	31
nds	Low German	15 139
ne	Nepali	124 961
new	Newari	4344
nl	Dutch	4 695 706
nn	Norwegian Nynorsk	5043
no	Norwegian	756 292
oc	Occitan	10 556
or	Odia	6138
os	Ossetic	8596
pa	Punjabi	25 879
pam	Pampanga	4
pl	Polish	5 686 688
pms	Piedmontese	7566
pnb	Western Panjabi	15 625
ps	Pashto	15 076
pt	Portuguese	7 611 586
qu	Quechua	1202
rm	Romansh	30
ro	Romanian	1 613 016
ru	Russian	31 972 436
rue	Rusyn	1
sa	Sanskrit	16 290
sah	Sakha	22 141
scn	Sicilian	21
sd	Sindhi	4366
		Continued

Code	Language	Doc. Count
sh	Serbian (Latin)	45 619
si	Sinhala	30 146
sk	Slovak	743 300
sl	Slovenian	293 415
so	Somali	39
sq	Albanian	208 223
sr	Serbian	162 126
su	Sundanese	1554
sv	Swedish	1 988 367
sw	Swahili	66 506
ta	Tamil	189 138
te	Telugu	72 914
tg	Tajik	19 353
th	Thai	838 422
tk	Turkmen	14 393
tl	Filipino	13 938
tr	Turkish	3 768 298
tt	Tatar	8724
tyv	Tuvinian	23
ug	Uyghur	47 035
uk	Ukrainian	1 789 621
ur	Urdu	110 291
uz	Uzbek	87 219
vec	Venetian	113
vi	Vietnamese	4 096 447
vls	West Flemish	1
vo	Volapük	6621
wa	Walloon	1383
war	Waray	23 687
wuu	Wu Chinese	222
xal	Kalmyk	51
xmf	Mingrelian	9706
yi	Yiddish	5646
yo	Yoruba	192
yue	Yue Chinese	3
zh	Chinese	8 744 984
Total Doc. Count		275 653 546

Table 8: List of languages included in the CulturaX dataset, along with the corresponding number of documents per language in the training sample used for LOLA. The language codes utilized are derived from the CulturaX dataset, which adheres to a combination of ISO 639-1 and ISO 639-3 standards. An exception is the use of *als*, which is considered obsolete; the ISO 639-3 standard designates *gsw* as its replacement.