P-MMEVAL: A Parallel Multilingual Multitask Benchmark for Consistent Evaluation of LLMs

Anonymous ACL submission

Abstract

Recent advancements in large language models (LLMs) showcase varied multilingual capabilities across tasks like translation, code generation, and reasoning. Previous assessments often limited their scope to fundamental natural language processing (NLP) or isolated capability-specific tasks. To alleviate this drawback, we aim to present a comprehensive multilingual multitask benchmark. First, we introduce P-MMEVAL, a large-scale benchmark covering fundamental and capability-specialized datasets. Furthermore, P-MMEVAL delivers consistent language coverage across various datasets and provides parallel samples. Finally, we conduct extensive experiments on representative multilingual model series to compare performances across models and tasks, explore the relationship between multilingual performances and factors such as tasks, model sizes, languages, and prompts, and examine the effectiveness of knowledge transfer from English to other languages. The resulting insights are intended to offer valuable guidance for future research [1].

1 Introduction

In recent years, large language models (LLMs, Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023; Bai et al., 2022, 2023) have raised significant interest in the artificial intelligence (AI) community. As most LLMs are English-centric, when we focus on the performances of a specific LLM, it generally refers to the evaluation results on English benchmarks. For example, early research focuses on reporting evaluation results on fundamental natural language processing (NLP) benchmarks. i.e, how accurately the LLM understands and generates text, including TRIVIAQA (Joshi et al., 2017a), WINOGRANDE (Sakaguchi et al., 2020), and HELLASWAG (Zellers et al., 2019). Nowadays, researchers are more interested in capability-specialized benchmarks, i.e., how well LLM performs on a group of specific task-solving problems, including GSM8K (Cobbe et al., 2021) for mathematical reasoning, MMLU (Hendrycks et al., 2021a) for knowledge acquisition, and HUMANEVAL (Chen et al., 2021) for code generation. However, there is currently little work on systematically evaluating the multilingual capabilities of LLMs. When developing and iterating LLMs, giving accurate and parallel evaluation results is crucial for identifying their multilingual capabilities and quantifying their performance.

Building a benchmark with both inclusive task coverage and strong linguistic parallelism is difficult. Measuring the multilingual abilities of a specific LLM, or comparing the quality of generated multilingual responses from one LLM to another, remains a big challenge in developing multilingual LLMs. Early work focuses on an isolated evaluation pipeline for a specific task, or to be more concrete, a specific perspective of LLM abilities: MHELLASWAG (Dac Lai et al., 2023) aims at collecting the multilingual understanding abilities, XLSUM (Hasan et al., 2021) mainly focus on evaluating the quality of generated multilingual text, HUMANEVAL-XL (Peng et al., 2024) is used for quantify how well-executed the generated code segments are, and MGSM (Shi et al., 2023) is made for testifying the performance on arithmetic reasoning. In modern research, for delivering simpler aggregation and comprehensive evaluation when judging model abilities, researchers collect several popular isolated benchmark tasks and propose a united, large-scale multilingual benchmark system like XTREME (Hu et al., 2020), XTREME-R (Ruder et al., 2021), XGLUE (Liang et al., 2020), MEGA (Ahuja et al., 2023), and BUFFET (Asai et al., 2024) for multi-task assessments. However, these large-scale benchmarks 1) are tailored predominantly to fundamental NLP tasks and 2) in-

[1] We will publish all the related resources in the future.

consistently cover multiple languages across their selected datasets.

In this paper, our goal is to develop a comprehensive multilingual multitask benchmark. To this end, we first include three datasets from fundamental NLP tasks covering both understanding and generation. The second phase of our endeavor involves a meticulous curation of the most intensely studied capability-specialized tasks in contemporary research including code generation, knowledge comprehension, mathematical reasoning, logical reasoning, and instruction following. Finally, we construct a collection of datasets P-MMEVAL, consisting of three fundamental NLP datasets and five advanced capability-specialized datasets. To maintain language coverage among all selected datasets, we unify 10 languages considering the cost and computational limitations via expert translation review to construct the missing multilingual portions.

To summarize, our contributions are as follows:

- We develop a multilingual multi-task benchmark P-MMEVAL that includes both fundamental and capability-specialized tasks, which ensures consistent language coverage across various datasets and provides parallel samples across different languages. This benchmark facilitates a thorough assessment of multilingual capabilities and enables unprecedented fairness and consistency in evaluating cross-lingual transfer capabilities.

- Our experiments offer a comprehensive analysis of the multilingual capabilities of various LLMs, showcasing performance across different prompts, models, languages, and tasks. Our analyses underscore a significant benchmark sensitivity in evaluating multilingual capabilities, indicating that the "nativeness" of the benchmark dramatically affects the observed multilingual evaluation results.

- We introduce the cross-lingual accuracy consistency ratio (CACR) to analyze the effectiveness of knowledge transfer from English to other languages across various target languages and task scenarios. Our analysis indicates that, among the tested tasks, code knowledge is the easiest to transfer, while logical reasoning proves the most difficult. Regarding specific languages, transfer is facilitated by linguistic similarity.

2 Related Work

Isolated Fundamental NLP Benchmarks Although diverse multilingual evaluation benchmarks have been established, they focused on basic language understanding and generation capabilities of models. Notable work includes XNLI (Conneau et al., 2018) for natural language inference, XCOPA (Ponti et al., 2020), MHELLASWAG (Dac Lai et al., 2023), and XWINOGRAD (Tikhonov and Ryabinin, 2021) for commonsense reasoning, PAWS-X (Yang et al., 2019) for paraphrase identification, XL-WIC (Raganato et al., 2020) for word sense disambiguation, as well as the span extraction QA datasets including XQUAD (Artetxe et al., 2020), MLQA (Lewis et al., 2020), and TYDIQA-GOLDP (Joshi et al., 2017b). Additional examples include XLSUM (Hasan et al., 2021) for text summarization and FLORES-200 (Costa-jussà et al., 2022) for machine translation. Each of those benchmarks is typically designed for a specific task, solely focusing on one aspect of the model's capabilities.

Unified Fundamental NLP Benchmarks There are also large-scale benchmarks that unify diverse existing datasets, aiming at offering a comprehensive evaluation of the model's abilities from various perspectives. For instance, XTREME (Hu et al., 2020) comprises four tasks related to natural language understanding (NLU). Its refined version, XTREME-R (Ruder et al., 2021), optimizes the specific datasets tailored for each task category within XTREME. The XGLUE (Liang et al., 2020), MEGA (Ahuja et al., 2023), and BUFFET (Asai et al., 2024) benchmarks integrate various datasets for both understanding and generation tasks.

Capability-specialized Multilingual Benchmarks The advanced task-solving capabilities of LLMs have garnered significant attention from the research community. The six capabilities that receive the most emphasis are mathematical reasoning (Cobbe et al., 2021; Hendrycks et al., 2021b), logical reasoning (Liu et al., 2020), instruction following (Li et al., 2023), knowledge comprehension (Hendrycks et al., 2021a), code generation (Chen et al., 2021), and conversational abilities (Bai et al., 2024). Typical multilingual benchmarks include MGSM (Shi et al., 2023) for mathematical reasoning, the OpenAI multilingual version of MMLU (MMMLU)[2] for knowledge

[2]https://huggingface.co/datasets/openai/MMMLU

Source	Task	Benchmarks	# Examples	Test sets	Metric
Existing	Generation	FLORES-200 (Costa-jussà et al., 2022)	1012 × 10	Annotation	BLEU
Extension	Understanding	XNLI (Conneau et al., 2018)	120 × 10 (3)	Translation	Acc
		MHELLASWAG (Dac Lai et al., 2023)	120 × 10 (3)	Translation	Acc
	Code generation	HUMANEVAL-XL (Peng et al., 2024)	80 × 10 (3) × 12	Translation	Pass@1
	Mathematical reasoning	MGSM (Shi et al., 2023)	250 × 10 (3)	Translation	Acc
	Logic reasoning	MLOGIQA (Liu et al., 2020)	80 × 10 (8)	Translation	Acc
	Knowledge	MMMLU (Hendrycks et al., 2021a)	400 × 10 (2)	Translation	Acc
	Instruction following	MIFEVAL (Zhou et al., 2023)	96 × 10 (9)	Translation	Acc

Table 1: An overview of the P-MMEVAL benchmark. In total, P-MMEVAL takes seven multilingual tasks into consideration, which is built on eight benchmarks. "# Examples" denotes "the number of examples per language" × "the number of involved languages" × "the number of programming languages" (special for HUMANEVAL-XL), and the numbers of extended languages are in parentheses. "Test sets" section describes the nature of the test sets (whether they are translations of English data or independently annotated).

comprehension, and HUMANEVAL-XL (Chen et al., 2021) for code generation.

All the benchmarks mentioned above focus either exclusively on fundamental NLP capabilities or on advanced application abilities. Additionally, there is inconsistent multilingual coverage across various datasets within a single multi-task benchmark. The proposed benchmark P-MMEVAL integrates three fundamental NLP datasets and five capability-specialized datasets, providing consistent language coverage across all selected datasets.

3 P-MMEval

We aim to build a comprehensive evaluation system that unifies diverse NLP and capability-specialized tasks, ensures consistent language coverage per task, and offers parallel samples across languages to facilitate consistent comparisons. The overview of our proposed P-MMEVAL is shown in Table 1.

3.1 Design Principles

Diversity in tasks First, the two key fundamental NLP tasks of generating and understanding are covered. More critically, through in-depth analysis, we identify and establish five kinds of core capabilities of current LLMs, including code generation, knowledge comprehension, mathematical reasoning, logical reasoning, and instruction following.

Diversity in languages To ensure that our benchmark can also help testify the cross-lingual transferability of LLMs, we unify 10 different languages spanning 7 language families, including English (*en*), Chinese (*zh*), Arabic (*ar*), Spanish (*es*), Japanese (*ja*), Korean (*ko*), Thai (*th*), French (*fr*), Portuguese (*pt*), and Vietnamese (*vi*).

3.2 Fundamental NLP Dataset Curation

In light of the diversity of fundamental NLP datasets, we meticulously select three datasets widely employed in research (Ahuja et al., 2023; Asai et al., 2024; Liang et al., 2020), spanning across the two major categories of understanding and generation. Below, we briefly summarize these three datasets.

i) XNLI: The natural language inference (NLI) dataset, XNLI (Conneau et al., 2018), involves classifying whether a hypothesis is entailed, contradicted, or unrelated to the premise.

ii) MHELLASWAG: The commonsense reasoning dataset MHELLASWAG (Zellers et al., 2019) consists of sentences or paragraphs, requiring models to predict the most likely option to complete the sentence or paragraph ending.

iii) FLORES200: The multilingual machine translation FLORES200 (Costa-jussà et al., 2022) is an evaluation benchmark for low-resource and multilingual machine translation.

3.3 Capability-specialized Dataset Curation

Besides the fundamental NLP tasks mentioned above, we also select one dataset for each of the five capability-specialized tasks. In detail, the involved specialized capabilities in P-MMEVAL are:

- **Code generation** We utilize HUMANEVAL-XL (Peng et al., 2024) dataset, which establishes connections between 23 natural languages (NLs) and 12 programming languages (PLs).
- **Mathematical reasoning** We use the MGSM (Shi et al., 2023) dataset, a multilingual ver-

Dataset	*zh*	*ar*	*es*	*ja*	*ko*	*th*	*fr*	*pt*	*vi*
XNLI	-	-	-	22.50	11.67	-	-	10.83	-
MHELLASWAG	-	-	-	82.50	77.50	26.67	-	-	-
HUMANEVAL-XL	-	-	-	42.50	23.75	31.25	-	-	-
MGSM	-	9.20	-	-	32.80	-	-	5.60	27.20
MLOGIQA	-	22.50	30.00	51.25	33.75	46.25	3.75	46.25	18.75
MMMLU	-	-	-	-	-	26.00	13.50	-	-
MIFEVAL	25.50	23.81	20.00	45.71	36.19	37.14	21.90	17.14	24.76

Table 2: The table presents the percentage of modifications made by professional translators to the machine translation results. The symbol "-" indicates that there are samples in the corresponding language and no translation construction is required.

sion translated from the monolingual GSM8K dataset consisting of math word problems.

- **Logical reasoning** We keep the original English and Chinese examples from origin LOGIQA (Liu et al., 2020) dataset.

- **Knowledge aqcuisition** We sample a subset of MMMLU comprising 200 "hard" samples and 200 "easy" samples. The performance of six diverse models (QWEN2.5-7B, QWEN2.5-72B, LLAMA3.1-8B, LLAMA3.1-70B, MISTRAL-NEMO, and MISTRAL-LARGE) is utilized as a proxy for selecting "hard" and "easy" samples. Concretely, we compile an "easy" subset comprising 6,335 instances where all models excel, and a "hard" subset consisting of 663 instances that challenge every model. Subsequently, guided by annotations from MMLU-REDUX (Gema et al., 2024), we refine these subsets by discarding 798 erroneous instances from the "easy" pool and 160 from the "hard" pool. Finally, we systematically sample 200 instances from each of the pruned pools, thus creating our finalized "easy" and "hard" evaluation sets.

- **Instruction following** We employ the English IFEVAL (Liu et al., 2020) dataset, which consists examples following pre-defined 25 types of "verifiable instruction".

3.4 Expansion of the Selected Datasets

To maintain consistency across all languages, we extend the support of some benchmark datasets on the missing languages by collecting human-annotated translation results. The number of expanded languages and samples for each dataset is listed in the "#Example" column of Table 1. More details of sampling are provided in Appendix Section A.

We initially generate translated examples using the advanced GPT-4o[3] model. Subsequently, a professional translation team conducts an exhaustive review of the machine translation outputs, correcting any errors, localizing vocabulary, and removing instances that do not translate well across languages. This meticulous process ensures both high translation quality and cultural adaptability.

The modification rate by post-review is detailed in Table 2. It is apparent that datasets contain translation errors to varying extents, with error rates peaking at 82.50%. This underscores the limitations of using raw machine-generated translations for dataset extension, highlighting the critical need for human review to maintain translation fidelity. Notably, among the most frequent errors are mistranslations of proper nouns and inconsistencies in terminology usage, followed by omissions. These trends indicate that the model currently struggles with specific domain terminology and maintaining contextual coherence.

3.5 Instruction selection

We utilize English instructions from OPENCOMPASS (Contributors, 2023) and LM-EVALUATION-HARNESS (Dac Lai et al., 2023). Among multiple instructions, we select a suitable one and make uniform modifications to ensure consistency across similar tasks. For zero-shot prompts, to increase the success rate of answer extraction, we add a constraint at the end of the instruction to some tasks, requiring the model to output the generated answers in a fixed format. In addition, we translate English instructions into multiple languages to construct native instructions.

4 Experiments

This section focuses on the following aspects: assessing the multilingual capabilities of different models; examining the influence of various

[3] gpt-4o-2024-05-13

Model	Understanding		Code generation	Mathematical reasoning	Logic reasoning	Knowledge	Instruction following	Generation	AVG_S	AVG_U
	XNLI	MHELLASWAG	HUMANEVAL-XL	MGSM	MLOGIQA	MMMLU	MIFEVAL	FLORES-200		
Open-source models (<7B)										
LLAMA3.2-1B	31.67	24.49	37.71	12.08	27.12	27.80	35.42	29.30	28.03	28.08
LLAMA3.2-3B	30.67	23.74	37.42	11.64	25.62	26.85	34.90	**36.85**	27.29	27.21
QWEN2.5-0.5B	22.25	19.68	33.92	13.12	14.62	30.25	30.21	15.95	24.42	20.97
QWEN2.5-1.5B	46.58	36.35	48.59	35.20	35.12	42.02	44.37	21.37	41.06	41.47
QWEN2.5-3B	60.08	48.09	60.75	69.40	39.38	46.27	66.46	25.75	**56.45**	**54.09**
GEMMA2-2B	53.50	45.31	51.54	44.52	34.88	40.85	56.67	24.00	45.69	49.41
Open-source models (7-14B)										
LLAMA3.1-8B	52.84	49.11	69.96	67.24	39.88	43.80	59.27	16.59	56.03	50.98
QWEN2.5-7B	67.17	62.92	71.88	81.08	45.88	49.83	77.71	32.76	65.28	65.05
GEMMA2-9B	57.92	65.62	69.96	81.28	41.50	49.23	79.17	**36.48**	64.23	61.77
MISTRAL-NEMO	54.25	55.73	57.38	76.52	41.75	44.88	60.00	33.65	56.11	54.99
QWEN2.5-14B	67.50	70.10	72.83	88.68	53.50	51.52	79.48	31.31	**69.20**	**68.80**
AYA-EXPANSE-8B	65.50	62.40	44.63	61.16	36.88	43.95	58.75	32.77	49.08	63.95
Open-source models (14-50B)										
QWEN2.5-32B	68.33	76.38	75.88	90.88	57.38	52.27	83.33	32.13	**71.95**	72.36
GEMMA2-27B	68.00	64.12	76.67	85.28	50.50	49.42	81.35	**42.23**	68.64	66.06
AYA-EXPANSE-32B	70.25	75.70	56.38	86.40	53.75	48.33	64.27	34.11	61.83	**72.98**
Open-source models (>50B)										
LLAMA3.1-70B	63.17	67.25	74.75	88.28	52.38	55.52	79.17	16.63	70.02	65.21
QWEN2.5-72B	71.42	75.95	76.00	91.00	58.38	52.67	87.60	41.55	**73.13**	**73.69**
MISTRAL-LARGE	69.58	69.04	77.17	90.48	53.50	51.85	83.23	**43.40**	71.25	69.31
Closed-source models										
GPT-4O	69.17	81.04	77.05	91.60	56.75	55.77	85.21	46.32	73.28	75.11
CLAUDE-3.7-SONNET	76.13	81.67	89.49	93.55	67.13	59.00	79.17	**48.18**	**77.49**	**78.90**

Table 3: Evaluation results of different models on P-MMEVAL. We gather those models by referring to their sizes. AVG_U and AVG_S represent the average score of the understanding and capability-specialized tasks, respectively. HUMANEVAL-XL score presents the average score of three programming languages.

prompts on multilingual performance; and comparing model performance in different languages.

4.1 Multilingual Models

We evaluate the performance of several representative instruction-tuned models – (i) closed-source models GPT-4O[4] (OpenAI, 2023) and CLAUDE-3.7-SONNET[5], (ii) open-source models including LLAMA3.1, LLAMA3.2 (Dubey et al., 2024), QWEN2.5 (Yang et al., 2024), MISTRAL-NEMO, MISTRAL-LARGE, GEMMA2, and AYA EXPANSE series (Dang et al., 2024).

4.2 Evaluation Settings

According to Zhao et al. (2021), the choice of prompts significantly impacts the evaluation results of LLMs and the model performance is sensitive to minor variations in prompting. In this study, we compare the evaluation results using the following prompts. EN: Instructions in English + input in the target language. Native: Instructions in the target language + input in the target language. EN-Few-Shot: Instructions in English + demonstrations in the target language + input in the target language.

For MGSM, we employ Chain of Thought (CoT) (Wei et al., 2022) reasoning, which guides the model to think step-by-step before providing a final answer. For the other datasets, direct answering is utilized, which requests the model to produce answers directly. The inference methods for these datasets align with the most commonly used settings. Notably, for MMMLU, we choose the prompt template following OpenAI `simple-evals` repository.[6] Specifically, CoT reasoning exhibits a significantly higher answer extraction failure rate compared to direct answering on small-sized LLMs (i.e., the number of parameters is less than 7B), leading to poor performance. Thus, we employ a direct answering prompt for small-sized LLMs.[7]

For the few-shot demonstrations, we primarily sample demonstrations from the validation set. For the missing multilingual portions, we utilize GPT-4O to translate these demonstrations from English into the missing languages.

4.3 Main Results

Table 3 presents an overview of the evaluation results. Unless otherwise noted, the standard EN prompt is applied to all datasets except FLORES-200, HUMANEVAL-XL, and MIFEVAL, where the Native prompt is required. The evaluation result

[4] `gpt-4o-2024-05-13`

[5] `claude-3-7-sonnet-20250219`

[6] https://github.com/openai/simple-evals

[7] The detailed evaluation prompts are illustrated in Appendix F.

on HUMANEVAL-XL is the average score across three programming languages including Python, JavaScript, and Java. See Appendix C for programming language evaluation details. For the Flores-200 dataset, in addition to reporting BLEU scores, we also provide COMET scores measured by wmt22-comet-da (Rei et al.) (see Appendix, Table 5).

First, the multilingual capabilities of models become stronger as the model sizes increase (Kaplan et al., 2020). One exception is that when the size of LLAMA3.2 increases from 1B to 3B, there is a slight decline in performance. The main reason for this is that LLAMA3.2-1B and LLAMA3.2-3B exhibit poor instruction-following capabilities, leading to a higher failure rate in answer extraction and, consequently, fluctuations in the final score. As the model size increases, the improvements in various multilingual tasks show significant differences. Evaluation results on the understanding and capability-specialized tasks show significant improvement in understanding context, processing semantic information, reasoning, and special abilities, with increasing model sizes. For example, for the QWEN2.5 series, the scores on the MGSM dataset for the 0.5B and 72B models are 13.12 and 91.00, respectively. In contrast, the models' performance on generation tasks is relatively weaker and shows slight improvement. Evaluations on the FLORES-200 datasets indicate that, despite the increase in model size, the generation capability does not improve proportionally. This may reflect the complexity of generating text that maintains logical coherence and contextual relevance, where increasing model sizes does not significantly enhance output quality.

In addition, QWEN2.5 demonstrates a strong multilingual performance on understanding and capability-specialized tasks, while GEMMA2 excels in generation tasks. Closed-source models GPT-4O and CLAUDE-3.7-SONNET generally outperform open-source models. The average performance gap between the best-performing open-source model and CLAUDE-3.7-SONNET reaches as high as 5.21%.

4.4 The Impact of Different Prompts on Model Performance

We explore three different prompting strategies: EN, Native, and En-Few-Shot. Table 4 illustrates the average performance of all evaluated open-source models on various datasets of P-MMEVAL.

Dataset	Native	EN	EN-Few-shot
MMMLU	44.30	44.69	45.70
MLOGIQA	42.27	41.96	44.88
MGSM	62.13	64.17	63.28
MHELLASWAG	52.03	53.37	59.07
XNLI	54.49	55.31	64.08
FLORES-200	30.00	24.31	29.18

Table 4: Comparison on P-MMEVAL using three different prompt settings.

Overall, the performance difference between the EN prompt and the Native prompt is minimal, remaining within 2%, indicating no substantial performance gap. However, in the case of the FLORES-200, the EN prompt results in a marked decline in performance compared to the Native prompt. We observe that models always generate responses in English when English instructions are used to describe the task for non-English data for generation tasks. On various datasets, the few-shot prompt leads to better model performance than the zero-shot prompt, as models achieve a higher success rate in extracting answers in the few-shot setting.

4.5 Language-Specific Model Performance Trends with Scale

We report the average performance per language on P-MMEval across various model sizes, excluding MMMLU, which is selected by models of different sizes, and FLORES-200, which excludes English performance. In addition, we do not consider models smaller than 7B, as their performance is often highly variable and sensitive to prompt phrasing.

As shown in Fig. 1, model performance varies by language, with English demonstrating the strongest capabilities, followed by Spanish and Portuguese. Thai has the poorest performance, followed by Japanese. Model performance in Thai is notably inferior to other languages, with a performance gap of up to 6.64% compared to Japanese. The distribution of training data and similarity between languages may explain these phenomena. Spanish and Portuguese are not only highly similar to English, but also have abundant language resources, reducing learning difficulty. In contrast, the Thai language has limited data resources, and Japanese belongs to an isolated language family. Another interesting phenomenon is that in Arabic, which is relatively resource-rich, smaller models around 7B perform nearly at the bottom, but when the model size increases to 14B and above, the performance significantly improves to a mid-to-low level. This indicates that Arabic has a complex linguistic sys-

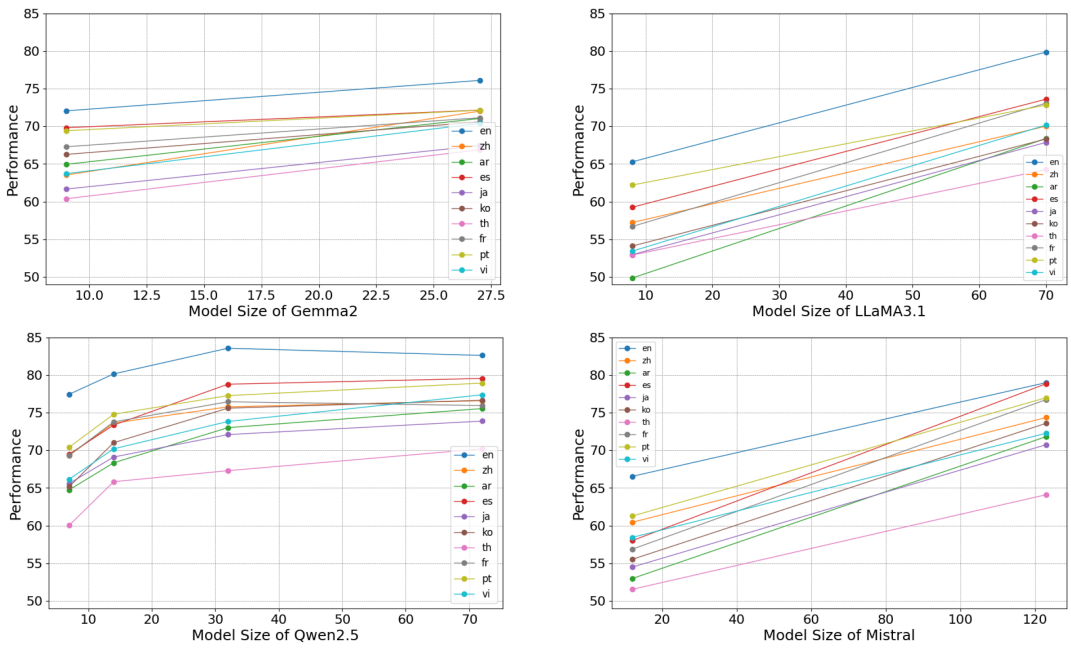

Figure 1: This figure illustrates model performance for each language.

tem that requires more parameters to capture its complexity and nuances.

Furthermore, we observe that in the Qwen series models (where Chinese data in the pre-training dataset is second only to English), the performance in Chinese is only mid-range, lagging behind Spanish and Portuguese. To investigate this apparent discrepancy, Appendix Section D provides a detailed comparison of multilingual capabilities assessed on benchmarks originating from English versus those from Chinese sources. This comparative analysis reveals that the same underlying multilingual ability of a model can yield disparate evaluation outcomes and exhibit different performance distributions when assessed using benchmarks derived from different source languages. These findings underscore a significant benchmark sensitivity in evaluating multilingual performance, indicating that the "nativeness" or origin of the benchmark dramatically affects the observed multilingual evaluation results.

5 Analysis of Cross-Lingual Transfer from English to Other Languages

To quantitatively evaluate the model's cross-lingual transfer success rate from English to target languages, we introduce the cross-lingual accuracy consistency ratio (CACR), computed over parallel multilingual test sets. This metric assesses the proportion of instances correctly predicted in English that are also correctly predicted in the target language. The metric is formally defined in Formula 1, where D_{en} and D_{tgt} denote aligned English and target language datasets, $f(\cdot)$ represents the model's prediction function.

5.1 Language-Specific Transfer Capabilities and the Influence of Benchmark Origin

We first examine the transfer success to various target languages based on benchmarks originating from English, and then compare these findings with results from a benchmark originating from Chinese to understand the impact of the benchmark's source language.

5.1.1 Transfer Performance on English-Origin Benchmarks

In Fig. 4, we report the average CACR for each language across the five tasks originating from English (MGSM, MMMLU, HUMANEVAL-XL, MHELLASWAG, and XNLI). We exclude the FLORES-200 and IFEVAL datasets, as they are not suitable for transfer analysis.

For all models, their CACR across all target languages also tends to improve as model size increases. This indicates that larger models typically possess stronger semantic representation learning and transfer capabilities.

In addition, the difficulty of transfer varies significantly across different target languages, with

$$CACR_{en->tgt} = \frac{\{x|x \in D_{en} \cap D_{tgt}, f(x_{en}) = y_{true} \wedge f(x_{tgt}) = y_{true}\}}{\{x|x \in D_{en}, f(x_{en}) = y_{true}\}}, \quad (1)$$

Romance languages like Spanish and Portuguese showing better transfer from English, while languages like Arabic present greater challenges. Linguistic characteristics (such as lexical and syntactic similarity to English) and the coverage of the language in pre-training data are among the factors that likely influence transfer effectiveness. These performance disparities also highlight the need for more targeted optimization and data augmentation for languages with low transfer success rates.

5.1.2 Impact of Benchmark Origin: English-Origin vs. Chinese-Origin

To investigate the influence of the original language of the benchmark on perceived transfer success, we compare the results from Fig. 4 (English-origin benchmarks) with those from Fig. 5, which reports the CACR transferring from English for each language on a task originating from Chinese (MLOGIQA).

When the benchmark originates from Chinese (Fig. 5), the CACR for transferring from English to Chinese is exceptionally high, often surpassing all other languages. In contrast, on English-origin benchmarks (Fig. 4), the CACR for Chinese, while respectable, is not as dominant. The impact of benchmark origin extends beyond just the Chinese language, leading to notable performance shifts for other languages as well. For instance, Portuguese, which demonstrates one of the highest CACR on English-origin benchmarks, sees its CACR drop to a mid-to-lower tier when the benchmark originates from Chinese. These indicate that the origin of the benchmark also affects the observed transfer success.

5.2 Comparison of the Difficulty of Transfer in Different Tasks

In Fig. 6, we report the average CACR for each task across all the nine languages included in P-MMEVAL. We exclude the FLORES-200 and IFEVAL datasets.

Model Scale Effect: For the GEMMA2, LLAMA3.1, and MISTRAL model series, the CACR generally shows an upward trend across all six evaluated tasks as model size (parameter count) increases. However, the QWEN2.5 model exhibits some differences. QWEN2.5 may achieve optimal transfer performance on certain tasks at a specific scale, with larger models potentially not yielding continued benefits or even encountering optimization bottlenecks.

Inter-task comparison: HUMANEVAL-XL (related to code generation/understanding) typically exhibits the highest CACR across all four models and various sizes. MGSM (mathematical reasoning) and MMMLU (knowledge understanding) are also consistently in the higher-performing tier, closely following HUMANEVAL-XL. The transfer performance of XNLI (natural language inference) is typically at an upper-mid level. MHELLASWAG (commonsense reasoning) generally performs at a lower-mid level. MLOGIQA (logical reasoning) is almost always at the lowest performance level across all models and sizes, indicating that this type of logical reasoning capability is the most challenging for cross-lingual transfer. This ranking of task difficulty shows high consistency across different model series.

Overall, increasing model size generally enhances the average cross-lingual transfer success rate, but this is not consistently effective for all models and all tasks, with QWEN2.5 showing transfer saturation on certain tasks. There are significant differences in the difficulty of cross-lingual transfer across tasks: code understanding and generation, mathematical reasoning, and knowledge understanding are relatively easier to transfer, while logical reasoning is the most challenging. This task difficulty hierarchy is largely consistent across different model series.

6 Conclusion

In this paper, we introduce a comprehensive multilingual multitask benchmark, P-MMEVAL, which covers both fundamental and capability-specialized tasks, ensuring consistent language coverage and providing parallel samples in multiple languages. Furthermore, we conduct extensive experiments on representative multilingual model series. These findings provide valuable guidance for future research, highlighting the importance of balanced and comprehensive training data, effective prompt engineering, and the need for targeted improvements in specific language capabilities.

Limitations

Through the above experiments and analyses, we summarize the following limitations:

1) Language Coverage: While P-MMEval currently covers 10 languages from 7 language families, there is a need to include more languages to better represent global linguistic diversity. Future work will focus on expanding the language coverage to ensure a more comprehensive evaluation of multilingual LLMs.

2) Task Diversity: P-MMEval includes eight representative tasks, but the rapidly evolving field of LLMs demands a broader range of tasks. Future work will focus on expanding the benchmark to cover more diverse and challenging tasks, providing a more thorough assessment of multilingual LLMs.

Ethics Statement

All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki Declaration and its later amendments or comparable ethical standards. This article does not contain any studies with animals performed by any of the authors. Informed consent was obtained from all individual participants included in the study.

References

Kabir Ahuja, Harshita Diddee, Rishav Hada, and et al. 2023. MEGA: multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4232–4267. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4623–4637. Association for Computational Linguistics.

Akari Asai, Sneha Kudugunta, Xinyan Yu, and et al. 2024. BUFFET: benchmarking large language models for few-shot cross-lingual transfer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 1771–1800. Association for Computational Linguistics.

Ge Bai, Jie Liu, Xingyuan Bu, and et al. 2024. Mt-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 7421–7454. Association for Computational Linguistics.

Jinze Bai, Shuai Bai, Yunfei Chu, and et al. 2023. Qwen technical report. *arXiv preprint arXiv:abs/2309.16609*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, and et al. 2022. Constitutional AI: harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*.

Tom B. Brown, Benjamin Mann, Nick Ryder, and et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Mark Chen, Jerry Tworek, Heewoo Jun, and et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:abs/2107.03374*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, and et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, and et al. 2018. XNLI: evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2475–2485. Association for Computational Linguistics.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

Marta R. Costa-jussà, James Cross, Onur Çelebi, and et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:abs/2207.04672*.

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.

John Dang, Shivalika Singh, Daniel D'souza, and et al. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:abs/2407.21783*.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, and et al. 2024. Are we done with mmlu? *arXiv preprint arXiv:abs/2406.04127*.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, and et al. 2021. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4693–4703. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, and et al. 2021a. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Dan Hendrycks, Collin Burns, Saurav Kadavath, and et al. 2021b. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, and et al. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *arXiv preprint arXiv:abs/2003.11080*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017a. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017b. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, and et al. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Patrick S. H. Lewis, Barlas Oguz, Ruty Rinott, and et al. 2020. MLQA: evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7315–7330. Association for Computational Linguistics.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Yaobo Liang, Nan Duan, Yeyun Gong, and et al. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6008–6018. Association for Computational Linguistics.

Jian Liu, Leyang Cui, Hanmeng Liu, and et al. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3622–3628. ijcai.org.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Qiwei Peng, Yekun Chai, and Xuhong Li. 2024. Humaneval-xl: A multilingual code generation benchmark for cross-lingual natural language generalization. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 8383–8394. ELRA and ICCL.

Edoardo Maria Ponti, Goran Glavas, Olga Majewska, and et al. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2362–2376. Association for Computational Linguistics.

Alessandro Raganato, Tommaso Pasini, José Camacho-Collados, and Mohammad Taher Pilehvar. 2020. Xl-wic: A multilingual benchmark for evaluating semantic contextualization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7193–7206. Association for Computational Linguistics.

Ricardo Rei, José G. C. de Souza, Duarte M. Alves, and et al. COMET-22: unbabel-ist 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation, WMT 2022, Abu Dhabi, United Arab Emirates (Hybrid), December 7-8, 2022*, pages 578–585.

Sebastian Ruder, Noah Constant, Jan A. Botha, and et al. 2021. XTREME-R: towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10215–10245. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI*

2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press.

Freda Shi, Mirac Suzgun, Markus Freitag, and et al. 2023. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

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, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 3534–3546. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, and et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, and et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

An Yang, Baosong Yang, Binyuan Hui, and et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:abs/2407.10671*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3685–3690. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4791–4800. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, and et al. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:abs/2311.07911*.

A Sampling Process for Each Dataset in P-MMEVAL

Specifically, since FLORES-200 already includes data for 10 languages, no additional translation was required. We retain the complete test set for evaluation.

For HUMANEVAL-XL and MGSM, which contain 80 and 250 examples per language, respectively, we ensured comprehensive coverage by translating the entire set for each language.

For single-task datasets XNLI, MHELLASWAG, and MLOGIQA, with large available test data, we follow established practices and select the first N examples for translation. This approach aligns with prior literature (Shi et al., 2023) and ensures consistency while managing computational and resource constraints.

For multi-task datasets such as MMMLU and IFEVAL, we adopt different strategies. For MMMLU, we sample a subset comprising 200 "hard" samples and 200 "easy" samples, by utilizing diverse model evaluation results as a proxy. For IFEVAL, we select 10 examples per task type, resulting in a total of 110 examples. During the translation verification process, 14 examples were removed due to quality issues, leaving a final set of 96 examples.

Model	COMET	BLEU
LLaMA3.2-1B	81.16	29.30
LLaMA3.2-3B	80.58	36.85
Qwen2.5-0.5B	80.06	15.95
Qwen2.5-1.5B	85.17	21.37
Qwen2.5-3B	87.08	25.75
Gemma2-2B	86.45	24.00
LLaMA3.1-8B	87.16	16.59
Qwen2.5-7B	87.62	32.76
Gemma2-9B	88.40	36.48
Mistral-Nemo	87.75	33.65
Qwen2.5-14B	87.26	31.31
Aya-expanse-8B	87.42	32.77
Qwen2.5-32B	88.56	32.13
Gemma2-27B	88.83	42.23
Aya-expanse-32B	88.61	34.11
LLaMA3.1-70B	88.27	16.63
Qwen2.5-72B	88.88	41.55
Mistral-Large	88.76	43.40

Table 5: The table displays the comparison between BLEU and COMET scores on the Flores-200 dataset.

B Evaluation of COMET Scores on the Flores-200 Dataset

In addition to the BLEU scores, we also provide COMET scores measured using the wmt22-comet-da model, shown in Table 5. For all tested models, the COMET scores are significantly higher than the BLEU scores, indicating that COMET is a more forgiving evaluation metric. Unlike BLEU, which requires strict literal matching, COMET focuses more on the semantics and fluency of the translation.

Additionally, COMET scores for all tested models are consistently high, generally ranging between 80 and 90, with negligible score differences observed between some models of large size gaps. This clustering of high scores and minimal variation indicates that COMET, in this specific evaluation scenario, likely lacked sufficient discriminative power to effectively measure nuanced performance differences between the various models or sizes. Consequently, we opt not to use COMET and continue to rely on BLEU as the primary evaluation metric for translation results, which, despite its own limitations, could still offer some relative performance insights in this context.

C Evaluation Results on Three Programming Languages of HumanEval-XL

Table 6 shows the evaluation results of all tested models on three programming languages of HumanEval-XL. Model performance in Python greatly exceeds the performance in the other two programming languages. For instance, Gemma2-2B scores 98.13 in Python, compared to 29.25 in JavaScript and 27.25 in Java. Additionally, as the model size increases, there is a noticeable improvement in performance for both JavaScript and Java.

D Comparison of the Multilingual Performance on Tasks originating from English and Chinese

On English-sourced benchmarks (Fig. 2), the model performs best in English, followed by excellent performance in Spanish and Portuguese (fellow Indo-European languages), and only moderate performance in Chinese. Conversely, on Chinese-sourced benchmarks (Fig. 3), the model performs best in Chinese. However, model performance in English fluctuates. On some models, such as Gemma2, it is only at a medium level.

	Python	JavaScript	Java
LLaMA3.2-1B	92.13	9.38	11.63
LLaMA3.2-3B	91.50	9.75	11.00
Qwen2.5-0.5B	78.38	14.25	9.13
Qwen2.5-1.5B	81.63	35.88	28.25
Qwen2.5-3B	84.00	53.75	44.50
Gemma2-2B	98.13	29.25	27.25
LLaMA3.1-8B	96.38	46.88	66.63
Qwen2.5-7B	86.75	68.00	60.88
Gemma2-9B	98.75	54.63	56.50
Mistral-Nemo	93.25	39.63	39.25
Qwen2.5-14B	84.50	72.75	61.25
Aya-expanse-8B	72.63	30.13	31.13
Qwen2.5-32B	89.38	73.13	65.13
Gemma2-27B	99.63	63.75	66.63
Aya-expanse-32B	96.25	39.00	33.88
LLaMA3.1-70B	98.75	63.38	62.13
Qwen2.5-72B	85.63	75.00	67.38
Mistral-Large	88.63	73.88	69.00
GPT-4o	89.13	77.88	64.13
Claude-3.7-sonnet	98.38	81.50	88.58

Table 6: The table presents the performance on three programming languages of HumanEval-XL.

Especially Portuguese, on Gemma, Mistral, and Qwen, the performance is below average. In addition, Japanese performance is among the lowest in English-sourced benchmarks, surpassing only Thai. However, performance improves to a mediocre level for most models on Chinese-sourced benchmarks. This difference may be due to lexical similarities between Japanese and Chinese. We suggest that when benchmarks are translated into other languages, the translation process itself, or inherent linguistic and cultural nuances, might inadvertently increase the difficulty for languages that are structurally and culturally more distant from the native languages.

E Analysis of Cross-Language Transfer from English to Other Languages

Fig. 4 and Fig. 5 illustrate the average CACR for each language on tasks originating from English and Chinese, respectively. Fig. 6 illustrates the average CACR across all nine languages for each task.

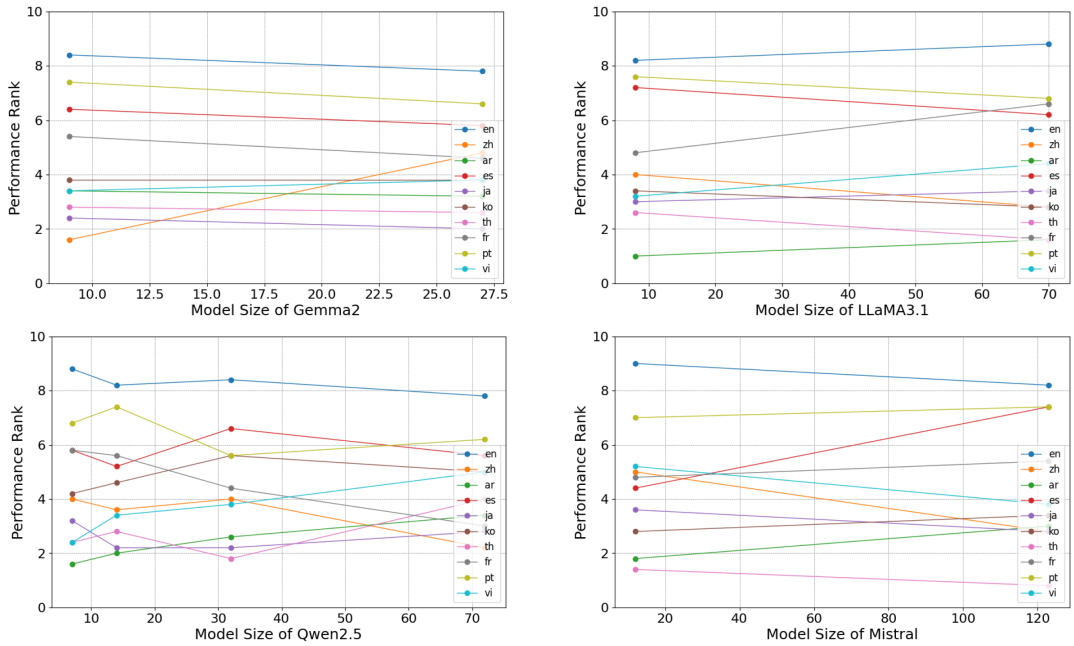

Figure 2: This figure presents the average performance rank for each language on English-sourced tasks, with higher ranks indicating better performance.

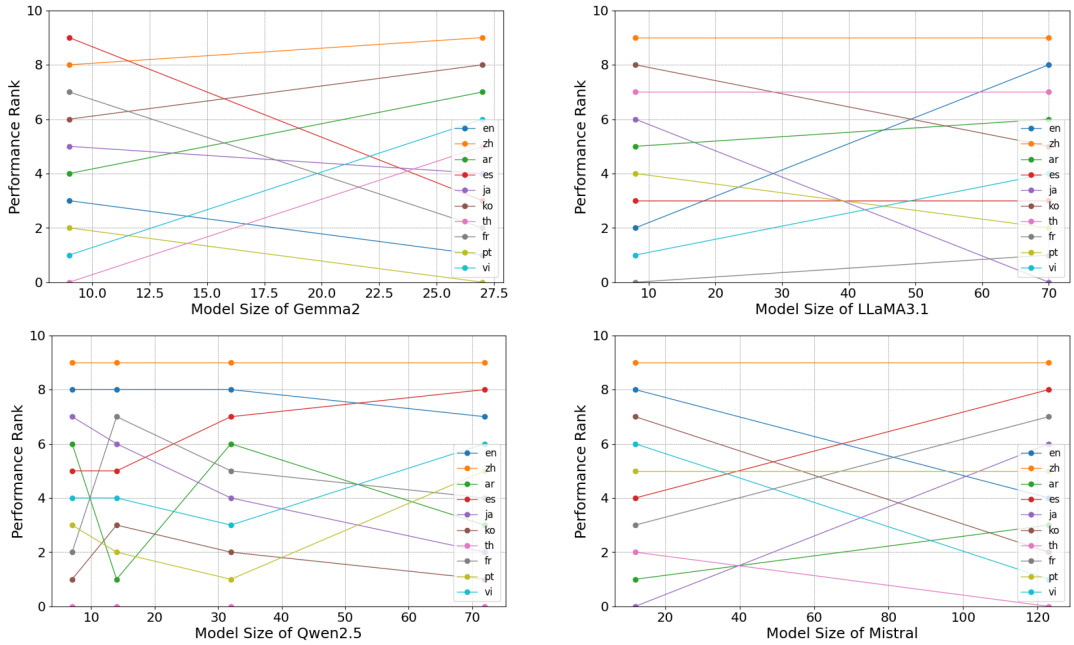

Figure 3: This figure shows the average performance rank for each language on Chinese-sourced tasks, with higher ranks indicating better performance.

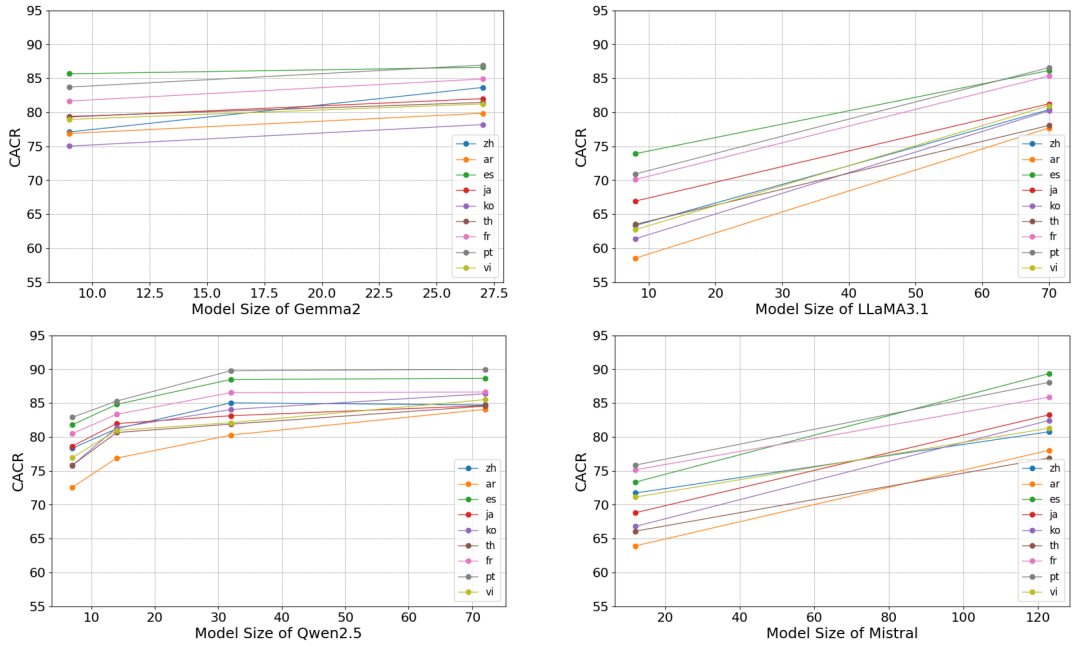

Figure 4: This figure illustrates the average CACR for each language on English-sourced tasks.

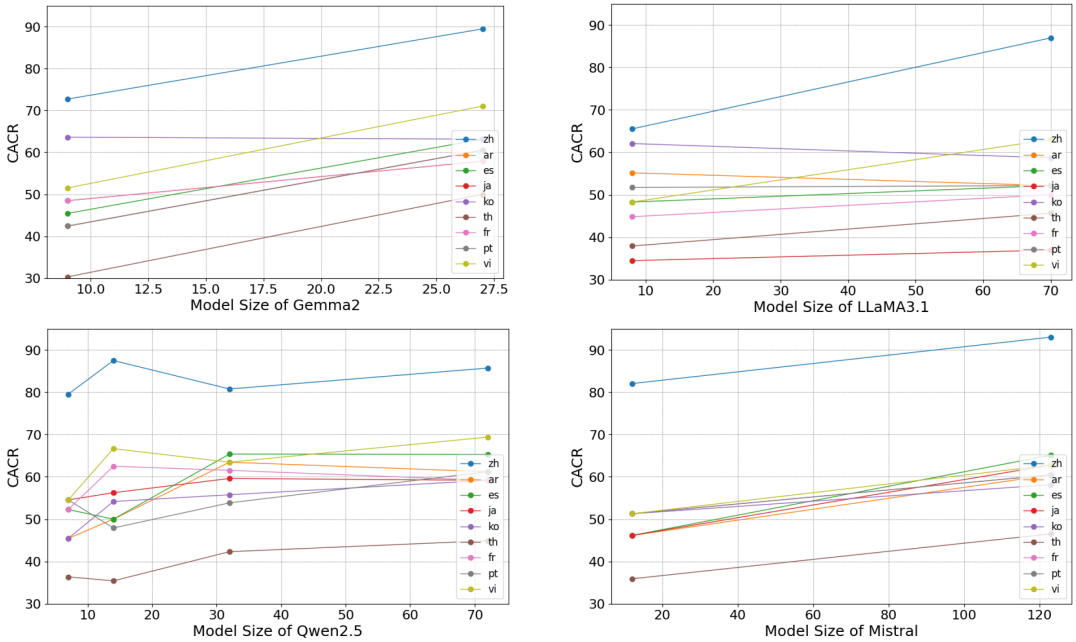

Figure 5: This figure illustrates the average CACR for each language on the MLogiQA task originating from Chinese.

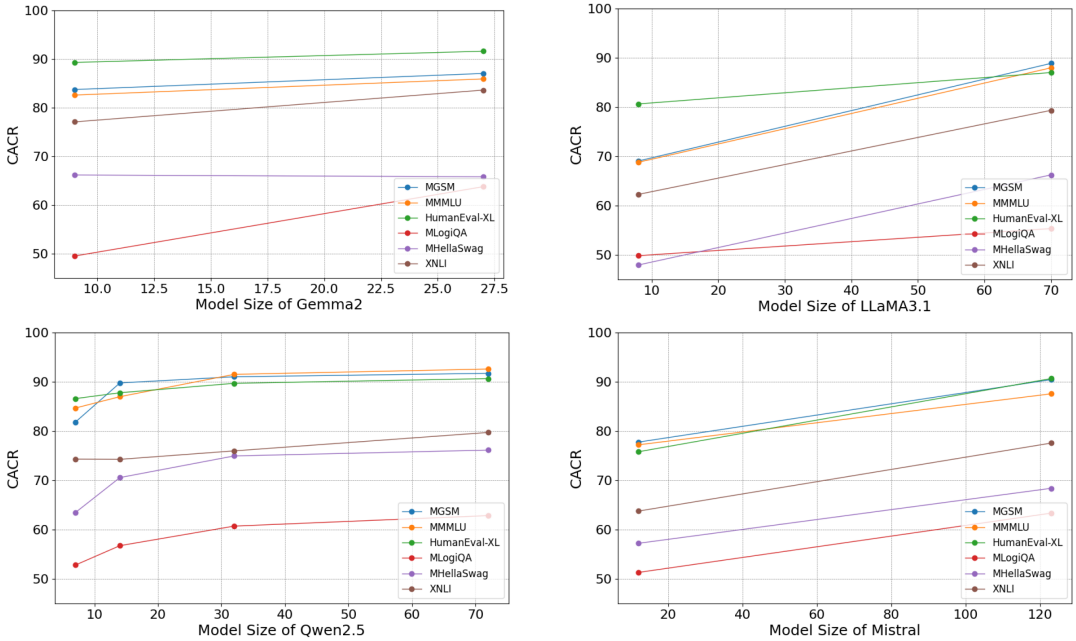

Figure 6: This figure displays the average CACR transferring from English to all target languages, broken down by task.

F The Prompt Utilized for Each Dataset

The section presents the inference prompt utilized for each dataset.

EN prompt for FLORES-200-en-x:
All: "Translate this sentence from English to {tgt_lang}.\n\n{src}\n"

Native prompt for FLORES-200-en-x:
zh: "将这个句子从英语翻译成中文。\n\n{src}"

th: "แปลประโยคนี้จากภาษาอังกฤษเป็นภาษาไทย.\n\n{src}"

ar: "ترجم هذه الجملة من الإنجليزية إلى العربية.\n\n{src}"

es: "Traduce esta oración del inglés al español.\n\n{src}"

ja: "この文を英語から日本語に翻訳してください。\n\n{src}"

ko: "이 문장을 영어에서 한국어로 번역하세요.\n\n{src}"

fr: "Traduisez cette phrase de l'anglais en français.\n\n{src}"

pt: "Traduza esta frase do inglês para o português.\n\n{src}"

vi: "Dịch câu này từ tiếng Anh sang tiếng Việt.\n\n{src}"

EN prompt for FLORES-x-en:
All: "Translate this sentence from {src_lang} to English.\n\n{src}\n"

Figure 7: This figure presents the prompt for the Flores-200 dataset.

EN prompt for MHELLASWAG:

All: "Input: {premise}\nOptions: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nPick the correct ending for the sentence from A, B, C, and D, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C or D."

Native prompt for MHELLASWAG:

zh: "输入：{premise}\n选项：\nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\n从 A, B, C 或者 D 中选出正确的句子结尾，并按照以下 JSON 格式返回：\n{'answer': '[choice]'}\n其中 [choice] 必须是 A, B, C 或者 D 其中之一。"

en: "Input: {premise}\nOptions: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nPick the correct ending for the sentence from A, B, C, and D, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C or D."

vi: "Nhập: {premise}\nLựa chọn: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nChọn kết thúc đúng cho câu từ A, B, C và D, và trả về theo định dạng JSON sau:\n{'answer': '[choice]'}\nTrong đó [choice] phải là một trong các A, B, C hoặc D."

th: "ข้อมูลนำเข้า: {premise}\nตัวเลือก: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nเลือกตอนจบที่ถูกต้องสำหรับประโยคจาก A, B, C และ D แล้วส่งคืนในรูปแบบ JSON ดังต่อไปนี้:\n{'answer': '[choice]'}\nโดย [choice] จะต้องเป็นหนึ่งใน A, B, C หรือ D."

ar: "الإدخال: {premise}\nالخيارات: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nاختر النهاية الصحيحة للجملة من A و B و C و D، وأعدها بتنسيق JSON التالي:\n{'answer': '[choice]'}\nحيث يجب أن يكون [choice] واحدًا من A أو B أو C أو D."

es: "Entrada: {premise}\nOpciones: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nElija el final correcto para la oración de A, B, C y D, y devuélvalo en el siguiente formato JSON:\n{'answer': '[choice]'}\ndonde [choice] debe ser uno de A, B, C o D."

ja: "入力: {premise}\n選択肢: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nA、B、C、Dから文の正しい結末を選び、次のJSON形式で返してください：\n{'answer': '[choice]'}\nここで、[choice]はA、B、C、またはDのいずれかでなければなりません。"

ko: "입력: {premise}\n옵션: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nA, B, C, D 중에서 문장의 올바른 엔딩을 선택하고, 다음 JSON 형식으로 반환하십시오:\n{'answer': '[choice]'}\n여기서 [choice]는 A, B, C 또는 D 중 하나여야 합니다."

fr: "Entrée : {premise}\nOptions : \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nChoisissez la fin correcte de la phrase parmi A, B, C et D, et renvoyez-la dans le format JSON suivant :\n{'answer': '[choice]'}\noù [choice] doit être l'un de A, B, C ou D."

pt: "Entrada: {premise}\nOpções: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nEscolha o final correto para a frase de A, B, C e D, e retorne-o no seguinte formato JSON:\n{'answer': '[choice]'}\nonde [choice] deve ser uma das opções A, B, C ou D."

Figure 8: This figure presents the prompt for the MHellaSwag dataset.

EN prompt for XNLI:
All: "Take the following as truth: {premise}\nThen the following statement: "{hypothesis}" is\nOptions: \nA. true\nB. inconclusive\nC. false\nSelect the correct option from A, B, and C, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, and C."

Native prompt for XNLI:
zh: "假设以下内容为真：{premise}\n考虑以下陈述：“{hypothesis}”\n该陈述是：\n选项：\nA. 真实的\nB. 无法确定\nC. 虚假的\n从 A, B 或者 C 中选择正确的选项，并按以下JSON格式返回：\n{'answer': '[choice]'}\n其中 [choice] 必须是 A, B 或者 C 其中之一。"

en: "Take the following as truth: {premise}\nThen the following statement: "{hypothesis}" is\nOptions: \nA. true\nB. inconclusive\nC. false\nSelect the correct option from A, B, and C, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, and C."

th: "ให้ถือว่าเป็นความจริง: {premise}\nแล้วข้อความต่อไปนี้: "{hypothesis}" เป็น\nตัวเลือก: \nA. จริง\nB. ไม่แน่นอน\nC. เท็จ\nเลือกตัวเลือกที่ถูกต้องจาก A, B, และ C และส่งคืนในรูปแบบ JSON ดังต่อไปนี้:\n{'answer': '[choice]'}\nโดยที่ [choice] ต้องเป็นหนึ่งใน A, B, และ C."

ar: "اعتبر ما يلي كحقيقة: {premise}\nثم الجملة التالية: "{hypothesis}" هي\nالخيارات: \nA. صحيحة\nB. غير صحيح\nC. خاطئة\nاختر الخيار الصحيح من A وB وC أدناه في تنسيق JSON التالي:\n{'answer': '[choice]'}\nحيث يجب أن يكون [choice] واحدًا من A أو B أو C."

es: "Tome lo siguiente como verdad: {premise}\nEntonces la siguiente afirmación: "{hypothesis}" es\nOpciones: \nA. verdadera\nB. inconclusa\nC. falsa\nSeleccione la opción correcta de A, B y C, y devuélvala en el siguiente formato JSON:\n{'answer': '[choice]'}\ndonde [choice] debe ser una de A, B y C."

ja: "次の内容を真実とみなしてください：{premise}\n次の文："{hypothesis}" は\n選択肢：\nA. 真\nB. 不確定\nC. 偽\nA、B、Cの中から正しい選択肢を選び、次のJSON形式で返してください：\n{'answer': '[choice]'}\nここで、[choice]はA、B、Cのいずれかでなければなりません。"

ko: "다음 내용을 진실로 간주하십시오: {premise}\n그렇다면 다음 진술: "{hypothesis}"는\n옵션: \nA. 사실\nB. 결론을 내릴 수 없음\nC. 거짓\nA, B, C 중에서 올바른 옵션을 선택하고 다음 JSON 형식으로 반환하십시오:\n{'answer': '[choice]'}\n여기서 [choice]는 A, B 및 C 중 하나여야 합니다."

fr: "Prenez ce qui suit comme vérité : {premise}\nAlors, l'affirmation suivante : "{hypothesis}" est\nOptions : \nA. vraie\nB. inconclusive\nC. fausse\nSélectionnez l'option correcte parmi A, B et C, puis renvoyez-la dans le format JSON suivant :\n{'answer': '[choice]'}\noù [choice] doit être l'un de A, B et C."

pt: "Considere o seguinte como verdade: {premise}\nEntão, a seguinte afirmação: "{hypothesis}" é\nOpções: \nA. verdadeira\nB. inconclusiva\nC. falsa\nSelecione a opção correta de A, B e C e retorne-a no seguinte formato JSON:\n{'answer': '[choice]'}\nonde [choice] deve ser uma das opções A, B ou C."

vi: "Xem điều sau đây là đúng: {premise}\nVậy tuyên bố sau đây: "{hypothesis}" là\nCác lựa chọn: \nA. đúng\nB. không kết luận\nC. sai\nChọn lựa chọn đúng từ A, B và C, và trả lại nó theo định dạng JSON sau:\n{'answer': '[choice]'}\ntrong đó [choice] phải là một trong A, B và C."

Figure 9: This figure presents the prompt for the XNLI dataset.

Native prompt for MGSM:

en: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "The answer is ". Do not add anything other than the integer answer after "The answer is ".\n\n{question}"

es: "Resuelve este problema matemático. Proporciona los pasos de razonamiento antes de dar la respuesta final en la última línea por sí misma en el formato de "La respuesta es ". No añadas nada más que la respuesta entera después de "La respuesta es ".\n\n{question}"

fr: "Résolvez ce problème de mathématiques. Donnez les étapes de raisonnement avant de fournir la réponse finale sur la dernière ligne elle-même dans le format de "La réponse est ". N'ajoutez rien d'autre que la réponse entière après "La réponse est ".\n\n{question}"

ja: "の数学の問題を解いてください。最終的な答えを出す前に、解答の推論過程を記述してください。そして最後の行には "答えは " の形式で答えを記述し、その後には整数の答え以外何も追加しないでください。\n\n{question}"

th: "แก้ปัญหาคณิตศาสตร์นี้ ให้ให้ขั้นตอนการใช้เหตุผลก่อนที่จะให้คำตอบสุดท้ายในบรรทัดสุดท้ายโดยอยู่ในรูปแบ "คำตอบคื " ไม่ควรเพิ่มอะไรนอกจากคำตอบที่เป็นจำนวนเต็มหลังจ "คำตอบคื "\n\n{question}"

zh: "解决这个数学问题。在最后一行给出答案前，请提供推理步骤。最后一行应该以 "答案是 " 的形式独立给出答案。在 "答案是 " 后不要添加除整数答案之外的任何内容。\n\n{question}"

ar: "قم بحل هذه المسألة الرياضية. قبل إعطاء الإجابة في السطر الأخير، يرجى تقديم خطوات الحل. يجب أن يتم تقديم الإجابة في السطر الأخير بشكل مستقل على شكل "الجواب هو ". لا تُضِف أي شيء بعد "الجواب هو " سوى العدد الصحيح للإجاب.\n\n{question}"

ko: "이 수학 문제를 해결하십시오. 마지막 줄에 답을 제시하기 전에 추론 단계를 제공하십시오. 마지막 줄은 "답변은 " 형식으로 독립적으로 답을 제시해야 합니다. "답변은 " 뒤에는 정수답 이외의 어떤 것도 추가하지 마십시오.\n\n{question}"

pt: "Resolva este problema matemático. Antes de dar a resposta na última linha, por favor, forneça os passos de raciocínio. A última linha deve apresentar a resposta de forma independente, começando com "A resposta é ". Após "A resposta é " não adicione nada além da resposta em número inteiro.\n\n{question}"

vi: "Giải quyết vấn đề toán học này. Trước khi đưa ra đáp án ở dòng cuối cùng, hãy cung cấp các bước lập luận. Dòng cuối cùng nên đưa ra đáp án dưới dạng "Câu trả lời là " một cách độc lập. Không thêm bất cứ nội dung nào ngoài đáp án là số nguyên sau "Câu trả lời là ".\n\n{question}"

Figure 10: This figure presents the Native prompt for the MGSM dataset.

EN prompt for MGSM:

en: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "The answer is ". Do not add anything other than the integer answer after "The answer is ".\n\n{question}"

es: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "La respuesta es ". Do not add anything other than the integer answer after "La respuesta es ".\n\n{question}"

fr: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "La réponse est ". Do not add anything other than the integer answer after "La réponse est ".\n\n{question}"

ja: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "答えは ". Do not add anything other than the integer answer after "答えは ".\n\n{question}"

th: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "คำตอบคือ ". Do not add anything other than the integer answer after "คำตอบคือ ".\n\n{question}"

zh: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "答案是 ". Do not add anything other than the integer answer after "答案是 ".\n\n{question}"

ar: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "وه باوجلا ". Do not add anything other than the integer answer after "وه باوجلا ".\n\n{question}"

ko: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "답변은 ". Do not add anything other than the integer answer after "답변은 ".\n\n{question}"

pt: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "A resposta é ". Do not add anything other than the integer answer after "A resposta é ".\n\n{question}"

vi: "Solve this math problem. Give the reasoning steps before giving the final answer on the last line by itself in the format of "Câu trả lời là ". Do not add anything other than the integer answer after "Câu trả lời là ".\n\n{question}"

Figure 11: This figure presents the EN prompt for the MGSM dataset.

EN prompt for MLOGIQA:
All: "Passage: {context}\nQuestion: {question}\nChoices:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nPlease choose the most suitable one among A, B, C and D as the answer to this question, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C and D."

Native prompt for MLOGIQA:
zh: "段落: {context}\n问题: {question}\n选择:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n请在 A、B、C 和 D 中选择最合适的一个作为此问题的答案，并以以下 JSON 格式返回：\n{'answer': '[choice]'}\n其中 [choice] 必须是 A、B、C 和 D 中的一项。"

en: "Passage: {context}\nQuestion: {question}\nChoices:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nPlease choose the most suitable one among A, B, C and D as the answer to this question, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C and D."

vi: "Đoạn văn: {context}\nCâu hỏi: {question}\nLựa chọn:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nVui lòng chọn câu trả lời phù hợp nhất trong số A, B, C và D cho câu hỏi này, và trả lại nó trong định dạng JSON sau:\n{'answer': '[choice]'}\ntrong đó [choice] phải là một trong A, B, C và D."

th: "ข้อความ: {context}\nคำถา: {question}\nตัวเลือก:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nโปรดเลือกข้อที่เหมาะสมที่สุดจาก A, B, C และ D เป็นคำตอบของคำถามน และส่งคืนในรูปแบบ JSON ดังต่อไปนี้:\n{'answer': '[choice]'}\nโดยที่ [choice] จะต้องเป็นหนึ่งใน A, B, C และ D."

ar: "الإدخال: {premise}\nالخيارات: \nA. {option_1}\nB. {option_2}\nC. {option_3}\nD. {option_4}\nاختر الإجابة الصحيحة للجملة من A و B و C و D، أو عدها بتنسيق JSON التالي:\n{'answer': '[choice]'}\nحيث يجب أن يكون [choice] واحداً من A أو B أو C أو D."

es: "Pasaje: {context}\nPregunta: {question}\nOpciones:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nPor favor, elija la más adecuada entre A, B, C y D como respuesta a esta pregunta, y devuélvala en el siguiente formato JSON:\n{'answer': '[choice]'}\ndonde [choice] debe ser uno de A, B, C y D."

ja: "本文: {context}\n質問: {question}\n選択肢:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nこの質問の答えとして A、B、C、D の中から最も適したものを選択し、次の JSON 形式で返してください：\n{'answer': '[choice]'}\nここで [choice] は A、B、C、または D のいずれかでなければなりません。"

ko: "구문: {context}\n질문: {question}\n선택:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n이 질문의 답으로 A, B, C 및 D 중 가장 적합한 것을 선택하고, 다음 JSON 형식으로 반환하십시오:\n{'answer': '[choice]'}\n여기서 [choice]는 A, B, C 및 D 중 하나여야 합니다."

fr: "Passage : {context}\nQuestion : {question}\nChoix :\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nVeuillez choisir le plus approprié parmi A, B, C et D comme réponse à cette question, et le renvoyer dans le format JSON suivant :\n{'answer': '[choice]'}\noù [choice] doit être l'un de A, B, C ou D."

pt: "Passagem: {context}\nPergunta: {question}\nOpções:\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\nPor favor, escolha a mais adequada entre A, B, C e D como resposta a esta pergunta, e retorne-a no seguinte formato JSON:\n{'answer': '[choice]'}\ndonde [choice] deve ser uma das opções A, B, C ou D."

Figure 12: This figure presents the prompt for the MLogiQA dataset.

EN prompt for MMMLU:
All: "The following is a multiple-choice question. Please choose the most suitable one among A, B, C and D as the answer to this question, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C and D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

Native prompt for MMMLU:
zh: "以下是一个多项选择题。请在 A、B、C 和 D 中选择最合适的一个作为此问题的答案，并以以下 JSON 格式返回：\n{'answer': '[choice]'}\n其中 [choice] 必须是 A、B、C 和 D 中的一项。\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

en: "The following is a multiple-choice question. Please choose the most suitable one among A, B, C and D as the answer to this question, and return it in the following JSON format:\n{'answer': '[choice]'}\nwhere [choice] must be one of A, B, C and D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

vi: "Dưới đây là một câu hỏi trắc nghiệm. Vui lòng chọn câu trả lời phù hợp nhất trong số A, B, C và D cho câu hỏi này, và trả lại nó trong định dạng JSON sau:\n{'answer': '[choice]'}\ntrong đó [choice] phải là một trong A, B, C và D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

th: "ต่อไปนี้คือคำถามแบบเลือกตอบหลายตัวเลือก โปรดเลือกข้อที่เหมาะสมที่สุดจาก A, B, C และ D เป็นคำตอบของคำถามนี้ และส่งคืนในรูปแบบ JSON ต่อไปนี้:\n{'answer': '[choice]'}\nโดยที่ [choice] จะต้องเป็นหนึ่งใน A, B, C และ D。\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

ar: "الأسئلة التالية هي أسئلة متعددة الخيارات. يرجى اختيار الإجابة الأنسب من بين A و B و C و D على هذه الأسئلة، وإعادتها بالتنسيق JSON التالي:\n{'answer': '[choice]'}\nحيث يكون [choice] أحد A و B و C و D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

es: "Lo siguiente es una pregunta de opción múltiple. Por favor, elija la más adecuada entre A, B, C y D como respuesta a esta pregunta, y devuélvala en el siguiente formato JSON:\n{'answer': '[choice]'}\ndonde [choice] debe ser uno de A, B, C y D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

ja: "以下は選択式の質問です。この質問の答えとして A、B、C、D の中から最も適したものを選択し、次の JSON 形式で返してください：\n{'answer': '[choice]'}\nここで [choice] は A、B、C、D のいずれかでなければなりません。\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

ko: "다음은 객관식 질문입니다. 이 질문의 답으로 A, B, C 및 D 중 가장 적합한 것을 선택하고 다음 JSON 형식으로 반환하십시오:\n{'answer': '[choice]'}\n여기서 [choice]는 A, B, C 및 D 중 하나여야 합니다.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

fr: "Ce qui suit est une question à choix multiple. Veuillez choisir la plus appropriée parmi A, B, C et D comme réponse à cette question, et la renvoyer dans le format JSON suivant :\n{'answer': '[choice]'}\noù [choice] doit être l'un de A, B, C ou D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

pt: "O seguinte é uma questão de múltipla escolha. Por favor, escolha a mais adequada entre A, B, C e D como resposta a esta pergunta, e retorne-a no seguinte formato JSON:\n{'answer': '[choice]'}\ndonde [choice] deve ser uma das opções A, B, C ou D.\n\n{question}\nA. {option_a}\nB. {option_b}\nC. {option_c}\nD. {option_d}\n"

Figure 13: This figure presents the prompt for the MMMLU dataset.