BenchMAX: A Comprehensive Multilingual Evaluation Suite for Large Language Models

Xu Huang [1] Wenhao Zhu [1] Hanxu Hu [4] Conghui He [2] Lei Li [3] Shujian Huang [1] Fei Yuan [2]

Abstract

Previous multilingual benchmarks focus primarily on simple understanding tasks, but for large language models (LLMs), we emphasize proficiency in instruction following, reasoning, long context understanding, code generation, and so on. However, measuring these advanced capabilities across languages is underexplored. To address the disparity, we introduce BenchMAX, a multi-way multilingual evaluation benchmark that allows for fair comparisons of these important abilities across languages. To maintain high quality, three distinct native-speaking annotators independently annotate each sample within all tasks after the data was machine-translated from English into 16 other languages. Additionally, we present a novel translation challenge stemming from dataset construction. Extensive experiments on BenchMAX reveal varying effectiveness of core capabilities across languages, highlighting performance gaps that cannot be bridged by simply scaling up model size. BenchMAX serves as a comprehensive multilingual evaluation platform, providing a promising test bed to promote the development of multilingual language models. The dataset[1] and code[2] are publicly accessible.
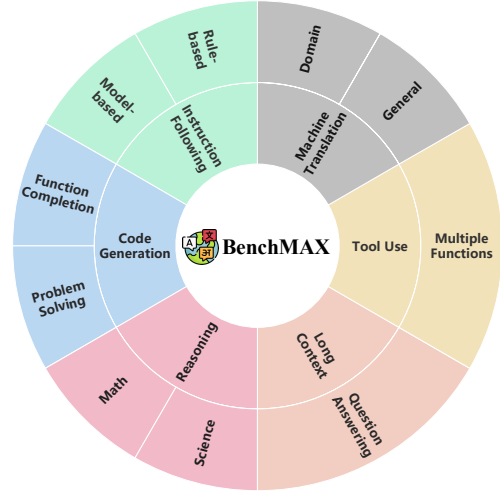
Figure 1. BenchMAX evaluates diverse advanced capabilities of LLMs in multilingual context.

standing, code generation, and so on (Ouyang et al., 2022; Cobbe et al., 2021; Su et al., 2024; Roziere et al., 2023; Lu et al., 2024; Sun et al., 2024). Inherently, these capabilities are language-agnostic. Consider a simple task like the acquisition of mathematical concept: the numerical outcome remains consistent regardless of whether one learns the arithmetic expression $1 + 1 = 2$ in English or Chinese. Similarly, when it comes to coding tasks, the choice between English or Chinese for articulating these instructions does not alter the fundamental logic of the code. However, numerous empirical studies have shown that LLMs' multilingual performance is quite unbalanced across different languages when handling same tasks (Shi et al., 2023; Zhu et al., 2024; Qi et al., 2023).

However, current benchmarks (Hendrycks et al., 2021; Lai et al., 2023; Singh et al., 2024; Wang et al., 2024a) do not support comprehensive testing of the language-agnostic abilities of LLMs, particularly in low-resource language settings, for several reasons. Tasks like XWinograd (Muennighoff et al., 2023) and XStoryCloze (Lin et al., 2022), based on multiple-choice formats, do not fully evaluate the generative capacities of LLMs. Additionally, the limited language overlap across existing benchmarks poses chal-

1. Introduction

Large language models (OpenAI et al., 2024; Gemini, 2024; DeepSeek-AI et al., 2024) have displayed remarkable proficiency across a wide range of tasks, mainly because they excel in instruction following, reasoning, long context under-

[1]National Key Laboratory for Novel Software Technology, Nanjing University [2]Shanghai Artificial Intelligence Laboratory [3]Carnegie Mellon University [4]University of Zurich. Correspondence to: Shujian Huang <huangsj@nju.edu.cn>, Fei Yuan <yuanfei@pjlab.org.cn>.

[1]https://huggingface.co/collections/ LLaMAX/benchmax-674d7a815a57baf97b5539f4
[2]https://github.com/CONE-MT/BenchMAX.git

Table 1. BenchMAX provides a more comprehensive analysis of LLM language-agnostic capabilities by covering a broader range of capability scenarios, language families, and script systems. # LG and # LG-Family denote the number of supported languages and the language families they belong to, respectively. $R_{non-Latin}$ refers to the proportion of languages that do not use the Latin script among supported languages. ∗ The results are from P-MMEval.

Tasks	Llama3.1 70B	Qwen2.5 72B	# LG	# LG-Family Diversity	$R_{non-Latin}$
XWinograd	69.7	83.7	6	3	50.0
XStoryCloze	70.3	83.6	13	11	38.5
MGSM*	88.3	79.2	10	7	50.0
MIFEVAL*	91.0	87.6	10	7	50.0
BenchMAX					
- Instruction Following	11.1	34.1			
- Code Generation	29.8	45.5	17	11	58.8
- Science Reasoning	35.8	39.4			
- Tool Use	44.3	61.8			

lenges in assessing LLM performance in specific languages. Recently, P-MMEval (Zhang et al., 2024) is proposed as a multilingual multitask benchmark, with the majority of its tasks still following a multiple-choice format. While it includes assessments like MGSM (Shi et al., 2023) and MIFEVAL that cover partial language-agnostic capabilities, LLMs exhibit remarkable performance, as shown in Table 1. This narrow focus leaves a significant gap between research evaluation and real-world applications.

To tackle this problem, we develop a comprehensive, multi-way, and challenging multilingual evaluation suite, called BenchMAX, to help the community better analyze and improve the language-agnostic capabilities of LLMs. Covering 17 languages[3], BenchMAX not only includes a broader range of language families but also emphasizes the diversity of writing systems across languages (Table 1). As demonstrated in Table 1, BenchMAX increases the percentage of studied languages that utilize the non-Latin script.

Meanwhile, BenchMAX highlights diverse language-agnostic advanced capabilities (Figure 1). We assess instruction following capability with rule-based (Zhou et al., 2023) and model-based (Li et al., 2024) evaluations, code generation capability in diverse scenarios (function-completion (Liu et al., 2024) / problem solving (Jain et al., 2024)), long context understanding capability (Hsieh et al., 2024), a verity of reasoning in math (Shi et al., 2023) and science (Rein et al., 2023), tool use (Srinivasan et al., 2023) in agent environments, and general (Costa-jussà et al., 2022) / domain translation. Domain translation, a by-product of data construction, poses a new challenge for LLM by necessitating fine-grained control and domain-specific terminology understanding over the translation process.

[3]The 17 languages include English, Spanish, French, German, Russian, Bengali, Japanese, Thai, Swahili, Chinese, Telugu, Arabic, Korean, Serbian, Czech, Hungarian, and Vietnamese.

To ensure high quality, we devise an annotation framework to optimize the dataset quality with human effort and LLM feedback. The process involves translating data from English to selected non-English languages using machine translation systems, post-editing each sample by three native-speaking annotators across all tasks, and picking the final translation version using a superior LLM that involves swapping sample positions for debiasing (Wang et al., 2024b; Li et al., 2024).

Leading multilingual LLMs are evaluated on BenchMAX, revealing that language significantly influences language-agnostic capabilities of existing LLMs. Interestingly, simply increasing the parameters can boost average performance on these tasks but does not universally reduce the performance gap across languages. Moreover, compared to general translation, domain translation not only poses new challenges for LLMs but also requires new evaluation metrics. The main contributions can be summarized as follows:

- We develop a comprehensive, multi-way multilingual benchmark across **17** languages for evaluating **6** crucial capabilities of LLMs on **10** diverse tasks.
- We propose a pipeline for curating high-quality mutlilingual datasets incorporating both human annotation and LLM-as-a-judge.
- Leading multilingual LLMs are evaluted on BenchMAX, and the related analyses provide a further understanding of the language-agnostic capabilities.

2. Related Work

Prior to the era of large language models, most multilingual benchmarks are designed to evaluate discriminative models and take the form of classification tasks, such as XNLI (Conneau et al., 2018), XCOPA (Ponti et al., 2020), XCSQA (Talmor et al., 2019), and so on (Lin et al., 2022; Muennighoff et al., 2023). However, due to their limited task complexity and the lack of diversity in format, these tasks become less practical. Recently, MGSM has become the most frequently used dataset in technical reports from leading LLM teams (Dubey et al., 2024; Gemini, 2024; OpenAI, 2024), which measures the mathematical reasoning capability across eleven languages. In this paper, we extend it to cover six additional diverse languages. Another widely used benchmark is the multilingual translated version of MMLU (Hendrycks et al., 2021; Lai et al., 2023; Singh et al., 2024), which assesses LLMs on knowledge-intensive tasks. However, due to the lack of a unified dataset version, scores are often difficult to compare across studies. Moreover, recent analyses have revealed that MMLU contains numerous ground truth errors (Gema et al., 2024), obscuring the accurate evaluation of LLM capabilities. To address these limitations, our work builds upon GPQA dataset (Rein et al., 2023) instead of MMLU, which offers higher-quality

Table 2. BenchMAX covers a wide range of language families and script systems.

Language	ISO	Language Family	Script System	Language	ISO	Language Family	Script System
Hungarian	hu	Uralic		Serbian	sr	Indo-European	Serbian Cyrillic
Vietnamese	vi	Austroasiatic		Korean	ko	Koreanic	Hangul / Chosŏn'gŭl
Spanish	es		Latin	Japanese	ja	Japonic	Mixed scripts of Chinese Characters and Hiragana, Katakana
Czech	cs	Indo-European		Arabic	ar	Afro-Asiatic	Arabic alphabet
French	fr			Thai	th	Kra–Dai	Thai
German	de			Swahili	sw	Niger–Congo	Latin
Russian	ru		Cyrillic	Chinese	zh	Sino-Tibetan	Chinese Characters
Bengali	bn		Bengali–Assamese	Telugu	te	Dravidian	Telugu

Table 3. Selection of core capabilities and details of task data. For IFEval, we filter out all language-specific instructions, thus remaining 429 samples. For Nexus, we only adopt the standardized_queries subset which contains 318 samples. For general translation datasets, the number of samples may vary in different translation directions, according to the number of parallel samples in TED and WMT24.

Capability	Category	Dataset	# Samples	Metric	Capability	Category	Dataset	# Samples	Metric
Instruction	Rule-based	IFeval	429	Accuracy	Code Generation	Function Completion	Humaneval+	164	Pass@1
Following	Model-based	Arena-hard	500	Win Rate		Problem Solving	LiveCodeBench_v4	713	
Reasoning	Math	MGSM	250	Exact Match	Translation	General	Flores+TED+WMT24	≥1012	spBLEU
	Science	GPQA	448			Domain	Annotated data above	2781	
Tool Use	Multiple Functions	Nexus	318	Accuracy	Long Context Modeling	Question Answering	RULER	800	Exact Match

annotations and poses greater challenges in domain-specific knowledge and reasoning evaluation. In addition to curating multilingual versions of MGSM and GPQA, we incorporate a broader range of capabilities, including long context modeling (Hsieh et al., 2024), tool use (Srinivasan et al., 2023), instruction following (Zhou et al., 2023), and more. This allows our benchmark to evaluate LLMs' multilingual capabilities more comprehensively compared to previous aggregated benchmarks, such as SeaEval (Wang et al., 2024a) and P-MMEval (Zhang et al., 2024). More importantly, all translations except the long context data in our benchmark are post-edited by native human experts after machine translation. This significantly improves both the quality and reliability of the dataset.

3. Benchmark Construction

In this section, we extend the evaluation of the core capabilities of LLMs into multilingual scenarios. To ensure sufficient linguistic diversity, we select 16 non-English languages (§ 3.1). Meanwhile, a diverse set of tasks designed to evaluate 6 crucial LLM capabilities is chosen to facilitate comprehensive assessment (§ 3.2). Subsequently, we introduce a rigorous pipeline (§ 3.3) that incorporates human annotators and LLMs to obtain a high-quality benchmark.

3.1. Language Selection

BenchMAX supports 17 selected languages to represent diverse language families and writing systems (Table 2).

3.2. Capabilities Selection

LLMs have demonstrated proficiency in understanding tasks such as text classification, sentiment analysis, and so on. However, their capabilities transcend text understanding, possessing the following intrinsic capabilities:

- **Instruction Following:** Following instructions capability is categorized into two distinct tasks based on evaluation paradigms: rule-based and model-based assessment.
- **Reasoning:** The capability to reason through intricate scenarios including both math reasoning and natural scientific (physics, chemistry, and biology) reasoning tasks.
- **Code Generation:** We primarily consider Python executable code generation in two settings, function completion and programming problem solving.
- **Long Context Modeling:** The ability to extract evidence from lengthy documents. We evaluate this capability through question-answering tasks with long documents (128k tokens).
- **Tool Use:** We assess the capability of utilizing tools effectively to correctly select and invoke a single function from multiple available functions based on given user queries.
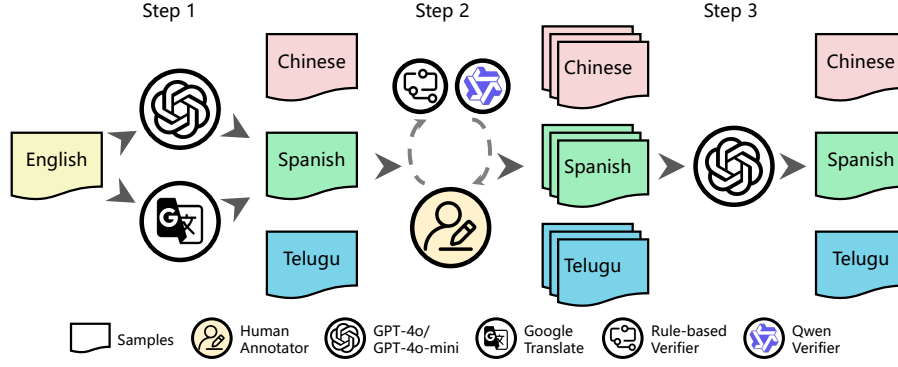- **Translation:** Translation involves accurately converting

Figure 2. The construction process of BenchMAX involves three steps: Step 1) translating data from English to non-English; Step 2) post-editing each sample by three human annotators; Step 3) selecting the final translation version.

Table 4. One example in rule-based instruction following task, which includes complex constraints. First, we enclose these constraints with special symbols and then use a machine translation system to translate from English to the target language. Finally, we reform the case structure by extracting the constraint from machine translation for human post-editing.

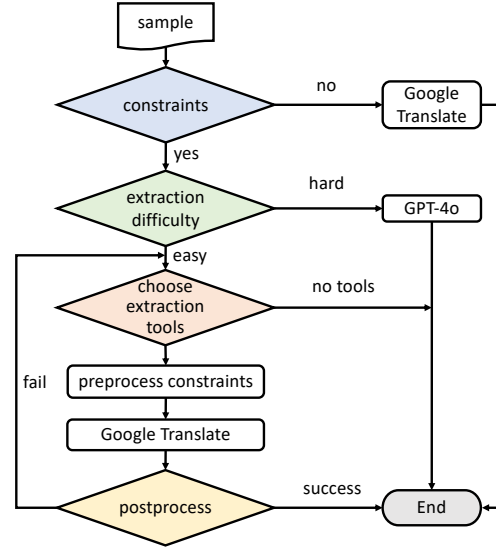| |
|---|
| **[Original Text]:** {prompt: Create an ad copy by expanding "Get 40 miles per gallon on the highway" in the form of a QA with a weird style. Your response should contain less than 8 sentences. Do not include keywords 'mileage' or 'fuel' in your response.
 instruction_id_list: ['length_constraints: number_sentences', 'keywords: forbidden_words']
 kwargs: [{'relation': 'less than', 'num_sentences': 8}, {'forbidden_words': ['mileage', 'fuel']}]}} |
| **[Translation Input]:** Create an ad copy by expanding "Get 40 miles per gallon on the highway" in the form of a QA with a weird style. Your response should contain less than 8 sentences. Do not include keywords '\<b\>mileage\</b\>' or '\<b\>fuel\</b\>' in your response. |
| **[Google Translation Result]:** 以风格怪异的问答形式扩展"在高速公路上每加仑行驶 40 英里"来创建广告文案。您的回复应少于 8 个句子。请勿在回复中包含关键字"\<b\>里程\</b\>"或"\<b\>燃料\</b\>"。 |
| **[Case Reform]** { prompt: 以风格怪异的问答形式扩展"在高速公路上每加仑行驶 40 英里"来创建广告文案。您的回复应少于 8 个句子。请勿在回复中包含关键字"里程"或"燃料"。
 instruction_id_list: ['length_constraints:number_sentences', 'keywords:forbidden_words']
 kwargs: [{'relation': 'less than', 'num_sentences': 8}, 'forbidden_words': ['里程', '燃料']] } |
| **[Human Post-Editing]** { "prompt": 以一种奇特风格的问答形式展开"在高速公路上每加仑行驶40英里"这句话，创建为一个广告文案。你的回复应该少于8句话。不要在你的回复中包含关键字"里程"或"燃料"。, <br> instruction_id_list: ['length_constraints:number_sentences', 'keywords:forbidden_words']
 kwargs: [{'relation': 'less than', 'num_sentences': 8}, 'forbidden_words': ['里程', '燃料']] } |

Figure 3. Flow chart illustrating the constraint extraction and pre-processing pipeline in the first step of our benchmark construction.

English by machines; 2) post-edit each sample by three native annotators; 3) pick the final translation version by GPT-4o-mini.

Step 1: Translating data from English to selected non-English languages by machine translation systems. We select between traditional translators such as Google Translate, and LLM-based ones like GPT-4o, depending on whether the task contains extractable constraints. As illustrated in Figure 3, if the task data contains constraints that are hard to extract, we prompt GPT-4o to translate the data and satisfy the constraints. Otherwise, we use Google Translate along with extraction tools. Extraction tools can include methods for extracting translated keywords by enclosing source keywords with special symbols, and for preserving source constraints by replacing constraints with placeholders before translation and restoring them afterwards.

text between languages while preserving semantic meaning. Beyond traditional translation tasks, we introduce the Domain Translation task, a by-product of the BenchMAX construction process. This task challenges models to translate specialized terminology and determine whether specific segments should be translated.

Further details on the datasets, sample sizes, and evaluation metrics are provided in Table 3. More detailed information can be found in Appendix A.

3.3. Construction

The way to obtain BenchMAX consists of three steps, as shown in Figure 2: 1) translate data from English to non-

Table 5. The recall rates using different groups of special symbols

Setting		Target Language			
		zh	es	fr	hu
w/o special symbols		0.68	0.68	0.68	0.68
symbol 1: 		0.91	0.89	0.88	0.93
symbol 2: ( )		0.88	0.91	0.89	0.92
symbol 3: ([ ])		0.82	0.89	0.87	0.92
Order 1	+ symbol 1	0.91	0.89	0.88	0.93
	+ symbol 2	0.93	0.93	0.90	0.95
Order 2	+ symbol 2	0.88	0.91	0.89	0.92
	+ symbol 1	0.90	0.93	0.90	0.95
	+ symbol 3	0.92	0.93	0.90	0.95

Taking an example of rule-based instruction following task as an example, as shown in Table 4, it requires extra processing to extract constraints from the translated instruction, as they are needed for verification. Inspired by Yuan et al. (2020), we enclose the keywords in the original instruction with special symbols, making them easy to extract from the translated result. As shown in Table 5, we explore various groups of special symbols and different orders and calculate the recall rates of keywords. Comparing to not using special symbols, apply any symbol group can greatly improve the recalls, while combining different symbol groups in multiple rounds can further improve the recalls. We choose *Order 1* as it can achieve better results with fewer groups than *Order 2*. The detailed recall results are in Appendix B.

Step 2: Post-editing each sample by three distinct native-speaking annotators in all tasks. To ensure high-quality dataset, we implement a multi-round annotation and verification process. 1) Each sample is given to three native-speaking annotators who are proficient in English and their native language. Considering the specialized nature of datasets like Science reasoning, annotators are required to hold at least a Bachelor's degree. 2) Two automatic verifiers - rule-based verifiers and model-based verifiers - are used to assess the quality of human annotation. Rule-based verifiers are used to ensure the satisfaction of constraints for certain tasks, such as the rule-based instruction following task. For model-based verifiers, we utilize the GEMBA-SQM prompt (Kocmi & Federmann, 2023) and employ Qwen2.5-72B-Instruct, a powerful multilingual model, estimating the quality of translations. Along with providing an overall score, the model offers detailed explanations of translation errors as feedback to annotators. Samples failing the rule-based verifier or scoring below a predefined threshold are identified as failed samples and refined in subsequent iterations. Each manually annotated dataset undergoes at least three iterations.

Step 3: Selecting the final translation version by LLMs. Initially, we ask a fourth annotator, uninvolved in the anno-

tation process, to choose the final version from the results revised by three individuals. Intriguingly, the selection by the fourth annotator exhibited a strong position bias, often favoring the initial annotation. This preference could be attributed to the uniformly high quality of the annotations, resulting in minimal discernible differences among them.

Debiasing for human annotators is costly in terms of both time and finance, because three translations encompass all permutations of six. Consequently, we employ GPT-4o-mini to select the final translation, as it is a powerful and balanced LLM across different languages. In particular, following Li et al. (2024), we adapt the LLM-Judge system instruction (see Appendix D) to suit pairwise translation evaluation. We firstly shuffle the positions of the three translations and conduct two battles to select a final version. In each battle between two translations, we perform two judgments by swapping their positions and determine one winner. The winner of the first two translations then battles against the third translation, determining the final winner.

4. Experimental Results

4.1. Evaluation Setup

Evaluated Models We focus primarily on multilingual post-trained models and evaluate both open-source and proprietary language models[4], including Llama3.1 (Dubey et al., 2024), Qwen2.5 (Qwen Team, 2024), Gemma2 (Team et al., 2024), InternLM2.5 (Cai et al., 2024), Aya-Expanse (Dang et al., 2024), DeepSeek-V3 (DeepSeek-AI et al., 2024), and GPT-4o-mini (OpenAI, 2024). Models detailed descriptions are in Appendix C.

Inference Configuration We adopt greedy decoding on most tasks except the problem solving task, on which the temperature is set to 0.2. The default chat template and system prompt of each model are applied. Detailed prompts are provided in Appendix D. Specifically, for instruction following tasks, the translated instructions are directly used as prompts. For reasoning tasks, we adopt the zero-shot native chain-of-thought (CoT) templates in LM-Evaluation-Harness (Gao et al., 2024). For other tasks, we use the original prompt templates provided in the corresponding repositories[5], and change the user inputs to other languages.

---

[4]Unless otherwise specified, all models discussed in this paper are post-trained versions.

[5]https://github.com/EleutherAI/lm-evaluation-harness
https://github.com/LiveCodeBench/LiveCodeBench
https://github.com/evalplus/evalplus
https://github.com/NVIDIA/RULER
https://github.com/lmarena/arena-hard-auto

Table 6. Performance comparison across models on BenchMAX tasks, averaged over 17 languages. Bold numbers indicate the best performance in each column. "Func Compl." refers to Function Completion, "Prob. Solving" to Problem Solving, and "Multi Func." to Multiple Functions scenarios where models must select and call one function from multiple options. Note that although DeepSeek-V3 supports 128 context, its API only supports 64K context, and we cannot deploy it on our server. Therefore, we have not evaluated its long-context capabilities yet.

| Model	Size	Instruction Following		Code Generation		Reasoning		Long Context	Tool Use	Translation			
										General		Domain	
		Rule-based	Model-based	Func Compl.	Prob. Solving	Math	Science	Question Answering	Multi Func.	En-X	X-En	En-X	X-En
InternLM2.5	7B	45.7	2.6	45.4	10.3	37.4	20.6	37.5	53.2	12.7	20.2	34.4	54.0
	20B	51.9	4.5	51.2	14.4	42.9	24.0	-	26.6	14.9	19.7	34.9	53.9
Aya-Expanse	8B	51.2	9.5	33.8	7.8	50.8	26.2	-	41.1	21.5	26.8	45.6	51.6
	32B	61.9	14.6	52.0	15.8	66.7	27.7	-	59.8	25.2	32.8	54.8	62.3
Gemma2	9B	63.0	9.1	53.9	16.6	72.0	23.9	-	61.4	27.2	33.2	57.5	61.9
	27B	62.4	16.0	66.7	24.6	75.3	26.7	-	64.7	30.4	34.5	64.8	66.2
Llama3.1	8B	62.6	5.1	52.9	14.1	63.4	23.8	68.3	45.0	24.6	29.8	53.9	62.9
	70B	76.2	11.1	69.7	29.8	79.7	35.8	57.4	44.3	31.1	**35.1**	64.5	**68.2**
Qwen2.5	7B	65.9	10.3	68.2	24.7	63.4	27.6	53.5	48.9	16.6	25.6	46.4	60.0
	72B	80.8	34.1	78.6	45.5	77.8	39.4	80.6	61.8	25.8	33.3	60.4	66.9
DeepSeek-V3	671B	**83.9**	**46.3**	**83.2**	**60.4**	**84.2**	**47.4**	-	69.2	**33.9**	34.5	**70.3**	67.8
GPT-4o-mini	-	79.1	23.6	78.7	37.0	76.9	34.1	**82.1**	**70.9**	30.3	33.9	67.7	67.6
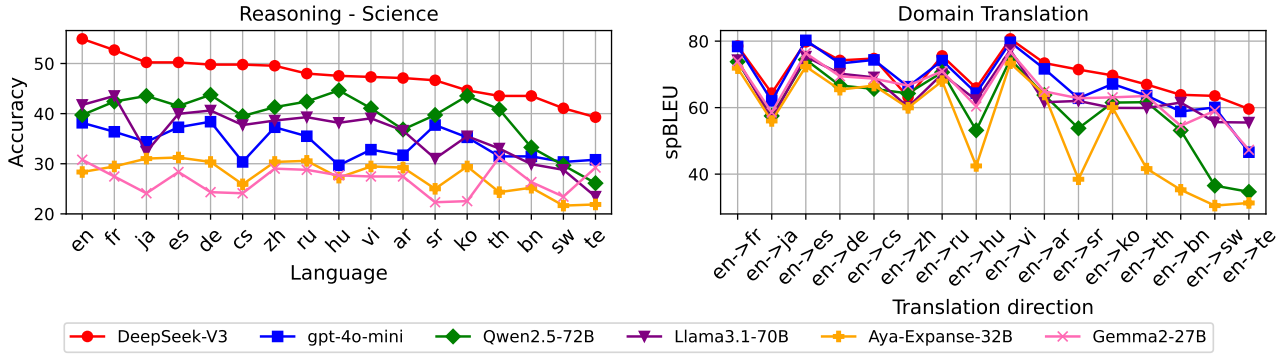
Figure 4. Performance of several models on the science reasoning task and the domain translation task across different languages. Models show unbalanced multilingual capabilities on these tasks.

4.2. Multilingual Benchmark Results

Table 6 shows the overall average performance of each model on each multilingual task. We take the top-performing ones and present their detailed performance on science reasoning and domain translation tasks, as illustrated in Figure 4. More detailed results are in Appendix E.

Model scaling improves overall multilingual performance while language disparities persist. As shown in Table 6, larger models consistently demonstrate enhanced multilingual capabilities across all domains, with few exceptions. However, the performance gap between English and non-English languages does not invariably diminish. We define GAP as the average performance gap between

English and other languages:

$$GAP = \frac{\sum_{l \neq \text{en}} \max(s(\text{en}) - s(l), 0)}{n - 1},$$

where $s(l)$ denotes the score on the task with language l, and n is the number of languages including English. As shown in Figure 5, when comparing models of different sizes, the proportion of larger models achieving smaller GAPs only slightly exceeds 0.5 for most model families except Qwen2.5. Gemma2-9B achieves smaller GAPs than Gemma2-27B in most tasks. These findings suggest that while scaling model size effectively improves overall multilingual performance, additional strategies may be needed to address the performance disparities across languages.
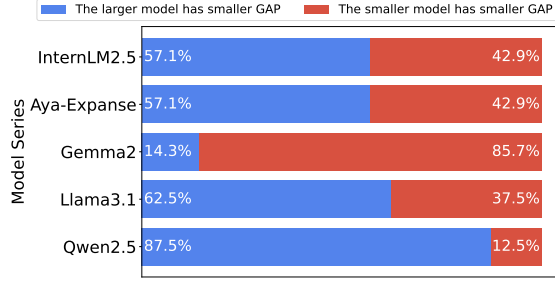
Figure 5. The proportion of tasks where larger models achieve a smaller GAP versus where smaller models perform better across different model families. larger models do not consistently have a smaller GAP, particularly in the Gemma2 family, where the smaller model outperforms the larger one in most cases.

Figure 6. F1 scores of correct answers in other languages to English across six tasks. Using English correctness as a reference, we treat other languages' correctness as predictions to measure their agreement with English. Results show high agreement between English and other languages on these tasks.

The effective utilization of language-agnostic capabilities remains challenging in multilingual contexts. Models' reasoning capabilities vary significantly across languages (Figure 4). While models generally achieve better performance in dominant languages like English, their performance in other languages often lags behind. This disparity can be attributed to the fact that multilingual task execution depends not only on language-agnostic reasoning but also on language-specific capabilities such as comprehension and generation. Therefore, when operating in languages where the model has weaker proficiency, it becomes difficult to fully leverage its language-agnostic capabilities. Interestingly, we observed some unexpected patterns where certain models excel in specific non-dominant languages compared to English. For example, Qwen2.5 demonstrates superior performance in Korean over English on the science reasoning task. This counter-intuitive phenomenon warrants further investigation in future work.

Model performance exhibits systematic bias towards high-resource languages, with Gemma2 as a notable exception. As shown in Figure 4, the performance curves of most models exhibit significant fluctuations across languages. High-resource languages such as French and Chinese consistently outperform low-resource languages like Telugu, Swahili, and Bengali. For instance, while DeepSeek-V3 achieves over 50% accuracy in science reasoning tasks for English and French, its performance drops notably to below 40% for Telugu. Our evaluation shows that almost all models have at least one language where they significantly underperform. This pattern can be partially attributed to development strategies - models like Aya-Expanse were not specifically optimized for the full range of languages in our evaluation. Notably, Gemma2 demonstrates unexpectedly balanced performance across most tasks (Figure 9), despite not being explicitly marketed as a multilingual model.
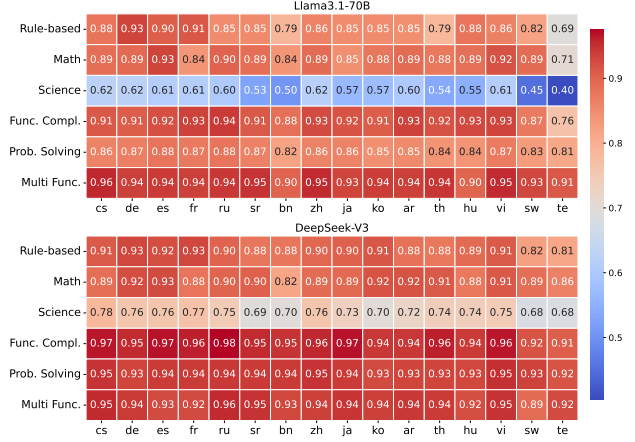
DeepSeek-V3 narrows the gap between open-source and closed-source models. As demonstrated in Table 6 and Figure 9, GPT-4o-mini demonstrates strong multilingual capabilities across various tasks. While its performance matches that of Qwen2.5-72B on several tasks, it falls short of DeepSeek-V3, the leading open-source model in our evaluation. This finding suggests that state-of-the-art open-source models are becoming increasingly competitive with their closed-source counterparts. Due to budget constraints, our evaluation is limited to GPT-4o-mini among closed-source models, and a more comprehensive comparison would be valuable for validating this trend.

Translation capabilities exhibit a positive correlation with other evaluated capabilities. We analyze the relationship between model's English-to-X translation capability and other capabilities using Spearman correlation coefficients (the right panel of Figure 7). When calculating correlations between domain-specific translation and task performance, we exclusively use data from the corresponding domains. The analysis reveals that domain-specific translation performance generally exhibits stronger correlations with task performance compared to general translation capabilities. A notable exception is that in the rule-based instruction-following task, we observe an inverse scaling effect: larger LLMs produce lower-quality translations compared to their smaller counterparts. This occurs because larger LLMs are more likely to execute instructions rather than strictly perform translation, known as prompt injection.

Models within the same family exhibit consistent performance patterns across languages. We calculate Spearman correlation coefficients to analyze the performance

Table 7. Different metrics are used to evaluate the En-X translation performance of selected models on specific domains.

Metric	Translation Model	Reasoning - Math				Reasoning - Science				Code generation - Prob. Solving			
		zh	de	sw	te	zh	de	sw	te	zh	de	sw	te
spBLEU	Gemma2-27B	40.0	51.4	38.2	29.2	80.6	84.8	66.2	57.5	85.5	78.5	76.2	52.3
	Llama3.1-70B	35.2	54.4	36.6	35.0	71.8	84.9	64.0	65.3	84.8	78.5	75.5	56.7
	Qwen2.5-72B	37.7	50.1	13.5	12.6	77.0	79.4	41.2	40.9	84.6	73.0	48.8	42.1
TER	Gemma2-27B	36.2	32.1	40.2	58.6	15.7	12.8	26.9	33.5	15.3	15.6	17.4	33.3
	Llama3.1-70B	36.0	30.1	44.0	51.8	19.6	12.9	28.4	26.9	15.1	15.4	17.9	46.8
	Qwen2.5-72B	33.1	33.5	76.8	85.7	15.4	16.8	65.8	51.4	14.3	19.6	38.3	53.5
xCOMET	Gemma2-27B	86.0	96.1	68.1	71.8	63.2	77.6	36.3	44.1	45.2	46.7	27.0	25.0
	Llama3.1-70B	86.8	95.6	66.1	74.0	63.7	77.4	37.1	46.8	43.5	46.3	27.8	28.9
	Qwen2.5-72B	87.6	95.6	24.5	30.1	65.2	76.3	20.3	28.4	45.0	45.7	18.0	18.4
Retention Rate	Gemma2-27B	1.00	1.08	0.98	0.99	0.98	0.98	1.07	0.81	1.02	0.96	0.89	0.95
	Llama3.1-70B	1.01	1.06	1.00	0.97	0.92	1.06	0.99	0.77	1.01	0.98	0.91	0.89
	Qwen2.5-72B	1.03	1.04	0.71	0.71	0.90	0.98	1.04	0.79	1.00	1.04	0.93	0.86
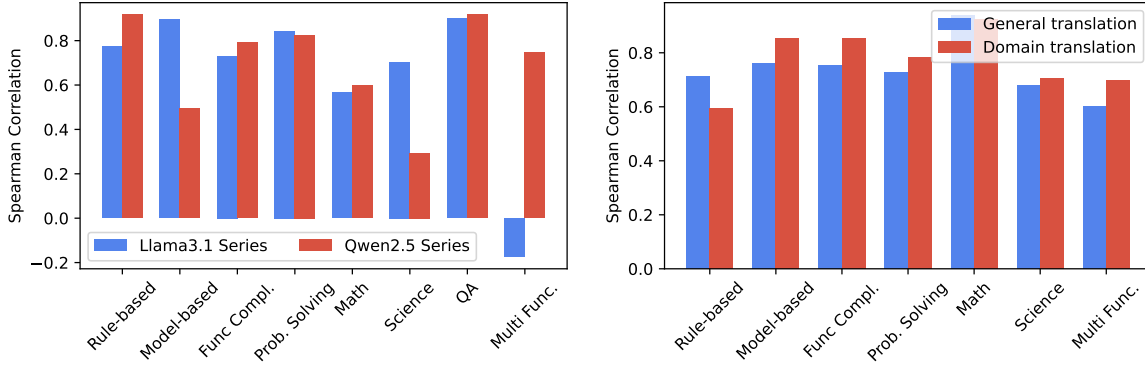
Figure 7. **Left:** Spearman Correlations between the performance of two models within the same family across different languages. **Right:** Spearman Correlations between the performance of models on general translation and domain translation.

similarity between models of the same family across different languages for each task. As shown in the left panel of Figure 7, models within the same family demonstrate strong correlations across various tasks, with most correlation coefficients exceeding 0.7. The exception is Llama3.1's negative correlation in tool use tasks.

5. Analysis

5.1. High correctness agreement between English and other languages

Although sometimes similar performance can be achieved across different languages for certain tasks, the specific problems being addressed may vary significantly. To examine this language alignment, we compute F1 scores measuring the agreement between problem-solving correctness in English versus other languages. Using English correctness as the reference labels, Figure 6 shows the F1 scores of Llama3.1-70B and DeepSeek-V3 on six subtasks of BenchMAX. Both these strong multilingual models demonstrate high correctness agreement on most tasks, with average F1

scores exceeding 0.9. F1 scores for low-resource languages are notably lower than those for high-resource languages. This pattern is particularly pronounced in science reasoning tasks, suggesting these knowledge-intensive problems pose unique challenges for cross-lingual knowledge transfer.

5.2. Challenges in evaluating domain-specific translation quality

Domain-specific texts often contain substantial content that does not require translation, which leads to inflated spBLEU scores. To address this limitation, we explore alternative evaluation approaches: the edit-distance metric TER (Snover et al., 2006), the model-based metric xCOMET-XXL (Guerreiro et al., 2024), and a novel performance retention rate that compares downstream task performance between model self-translation and human translations. Table 7 presents these metric scores across selected tasks and languages. Traditional metrics prove unreliable for domain-specific translation evaluation. Both spBLEU and TER show extreme values in Science and Programming tasks due to large portions of unchanged text, fail-
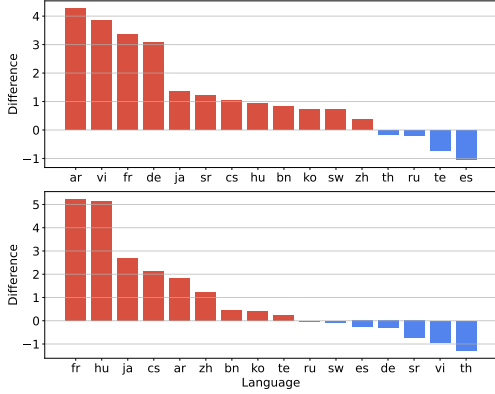
Figure 8. The performance differences between using human-translated data and machine-translated data. The upper figure depicts the performance difference of Llama3.1-70B on the rule-based instruction-following task, while the lower figure presents the performance difference for Qwen2.5-72B on the same task.

ing to capture the quality of crucial translated segments. The model-based metric xCOMET shows high variance across different scenarios, particularly struggling with low-resource languages and specialized domains. Moreover, the performance retention rate doesn't work as expected as many values are greater than 1, which fails to evaluate translations subtly. These findings highlight the need for specialized metrics for domain-specific translation evaluation, which we identify as an important direction for future research.

5.3. Machine-translated task data leads to inconsistent assessment of LLM capabilities

Evaluating models on machine-translated data can lead to both overestimation and underestimation of model performance. This inconsistency is evidenced by the retention rates exceeding 1.0 in Table 7, suggesting occasional overestimation. However, underestimation is more prevalent (Figure 8). Both Llama3.1-70B and Qwen2.5-72B achieve significantly higher scores on human-translated rule-based instruction following tasks across several languages, including French, Arabic, and Hungarian, with performance gaps exceeding 4 points in some cases.

5.4. Conventional understanding tasks are inadequate to evaluate multilingual capabilities of LLMs

Prior work like Aya-Expanse (Dang et al., 2024) relies on conventional understanding tasks such as XCOPA and XWinograd for multilingual evaluation. On these metrics, Gemma2-9B achieves the best performance, followed by Aya-Expanse-8B, Llama3.1-8B, and Qwen2.5-7B. However, our evaluation through BenchMAX reveals a different pattern: Qwen2.5-7B demonstrates superior multilingual capabilities on generation tasks, while Aya-Expanse mod-

Table 8. Ranks of four models in our tasks compared to discriminative tasks. The rankings of Discriminative tasks are from Dang et al. (2024).

Model	Rule-based	Func Compl.	Discriminative
Qwen2.5-7B	No.1	No.1	No.4
Llama3.1-8B	No.3	No.3	No.3
Aya-Expanse-8B	No.4	No.4	No.2
Gemma2-9B	No.2	No.2	No.1

els show notably weaker performance on code generation tasks, as shown in Table 8. This discrepancy highlights the importance of comprehensive evaluation frameworks that incorporate both understanding and generation tasks to accurately assess the multilingual capabilities of LLMs.

6. Conclusion

In this work, we introduce BenchMAX, a comprehensive, high-quality, and multiway parallel multilingual benchmark comprising ten tasks designed to assess crucial capabilities across seventeen diverse languages. The multilingual task data is initially translated from English using machine translation and subsequently refined through multiple iterations of post-editing by native speakers, ensuring high data quality. Through extensive model evaluations, we find that the language-agnostic capabilities of state-of-the-art LLMs remain uneven across different languages. While increasing model size consistently enhances multilingual performance, the performance gap between English and other languages persists, highlighting the need for further efforts to achieve balanced multilingual capabilities.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Artetxe, M., Ruder, S., and Yogatama, D. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020. URL https://aclanthology.org/2020.acl-main.421.

Cai, Z., Cao, M., Chen, H., Chen, K., Chen, K., Chen, X., Chen, X., Chen, Z., Chen, Z., Chu, P., et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024. URL https://arxiv.org/abs/2403.17297.

Cettolo, M., Girardi, C., and Federico, M. WIT3: Web inventory of transcribed and translated talks. In *Pro-*

ceedings of the 16th Annual conference of the European Association for Machine Translation, pp. 261–268, Trento, Italy, May 28–30 2012. European Association for Machine Translation. URL https://www.aclweb.org/anthology/2012.eamt-1.60.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/pdf/2107.03374.

Chiang, W., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhu, B., Zhang, H., Jordan, M. I., Gonzalez, J. E., and Stoica, I. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=3MW8GKNyzI.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/pdf/2110.14168.

Conneau, A., Rinott, R., Lample, G., Williams, A., Bowman, S., Schwenk, H., and Stoyanov, V. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018. URL https://aclanthology.org/D18-1269.

Costa-jussà, M. R., Cross, J., Çelebi, O., Elbayad, M., Heafield, K., Heffernan, K., Kalbassi, E., Lam, J., Licht, D., Maillard, J., et al. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*, 2022.

Dang, J., Singh, S., D'souza, D., Ahmadian, A., Salamanca, A., Smith, M., Peppin, A., Hong, S., Govindassamy, M., Zhao, T., et al. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*, 2024. URL https://arxiv.org/abs/2412.04261.

DeepSeek-AI, Liu, A., Feng, B., Xue, B., Wang, B.-L., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Guo, D., Yang, D., Chen, D., Ji, D.-L., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Zhang, H., Ding, H., Xin, H., Gao, H., Li, H., Qu, H., Cai, J. L., Liang, J., Guo, J., Ni, J., Li, J., Wang, J., Chen, J., Chen, J., Yuan, J., Qiu, J., Li, J., Song, J.-M., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Xu, L., Xia, L., Zhao, L., Wang, L., Zhang, L., Li, M., Wang, M., Zhang, M., Zhang, M., Tang, M., Li, M., Tian, N., Huang, P., Wang, P., Zhang, P., Wang, Q., Zhu, Q., Chen, Q., Du, Q., Chen, R. J., Jin, R. L., Ge, R., Zhang, R., Pan, R., Wang, R., Xu, R., Zhang, R., Chen, R., Li, S. S., Lu, S., Zhou, S., Chen, S., Wu, S.-P., Ye, S., Ma, S., Wang, S., Zhou, S., Yu, S., Zhou, S., Pan, S., Wang, T., Yun, T., Pei, T., Sun, T., Xiao, W. L., Zeng, W., Zhao, W., An, W., Liu, W., Liang, W., Gao, W., Yu, W.-X., Zhang, W., Li, X. Q., Jin, X., Wang, X., Bi, X., Liu, X., Wang, X., Shen, X.-C., Chen, X., Zhang, X., Chen, X., Nie, X., Sun, X., Wang, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yu, X., Song, X., Shan, X., Zhou, X., Yang, X., Li, X., Su, X., Lin, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhu, Y. X., Zhang, Y., Xu, Y., Huang, Y., Li, Y., Zhao, Y., Sun, Y., Li, Y., Wang, Y., Yu, Y., Zheng, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Tang, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y.-B., Liu, Y., Guo, Y., Wu, Y., Ou, Y., Zhu, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Zha, Y., Xiong, Y., Ma, Y., Yan, Y., Luo, Y.-W., mei You, Y., Liu, Y., Zhou, Y., Wu, Z. F., Ren, Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Huang, Z., Zhang, Z., Xie, Z., guo Zhang, Z., Hao, Z., Gou, Z., Ma, Z., Yan, Z., Shao, Z., Xu, Z., Wu, Z., Zhang, Z., Li, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z.-A., Xie, Z., Song, Z., Gao, Z., and Pan, Z. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. URL https://arxiv.org/pdf/2412.19437.

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/pdf/2407.21783.

Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., Le Noac'h, A., Li, H., McDonell, K., Muennighoff, N., Ociepa, C., Phang, J., Reynolds, L., Schoelkopf, H., Skowron, A., Sutawika, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. A framework for few-shot language model evaluation, 07 2024. URL https://zenodo.org/records/12608602.

Gema, A. P., Leang, J. O. J., Hong, G., Devoto, A., Mancino, A. C. M., Saxena, R., He, X., Zhao, Y., Du, X., Madani, M. R. G., Barale, C., McHardy, R., Harris, J., Kaddour, J., van Krieken, E., and Minervini, P. Are we done with mmlu?, 2024. URL https://arxiv.org/abs/2406.04127.

Gemini. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. URL https://arxiv.org/pdf/2403.05530.

Guerreiro, N. M., Rei, R., Stigt, D. v., Coheur, L., Colombo, P., and Martins, A. F. xcomet: Transparent machine

translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995, 2024. URL https://doi.org/10.1162/tacl_a_00683.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/pdf?id=d7KBjmI3GmQ.

Hsieh, C.-P., Sun, S., Kriman, S., Acharya, S., Rekesh, D., Jia, F., Zhang, Y., and Ginsburg, B. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*, 2024. URL https://arxiv.org/pdf/2404.06654.

Jain, N., Han, K., Gu, A., Li, W.-D., Yan, F., Zhang, T., Wang, S., Solar-Lezama, A., Sen, K., and Stoica, I. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024. URL https://arxiv.org/pdf/2403.07974.

Kocmi, T. and Federmann, C. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pp. 193–203, Tampere, Finland, June 2023. European Association for Machine Translation. URL https://aclanthology.org/2023.eamt-1.19/.

Kocmi, T., Avramidis, E., Bawden, R., Bojar, O., Dvorkovich, A., Federmann, C., Fishel, M., Freitag, M., Gowda, T., Grundkiewicz, R., et al. Findings of the wmt24 general machine translation shared task: the llm era is here but mt is not solved yet. In *Proceedings of the Ninth Conference on Machine Translation*, pp. 1–46, 2024. URL https://www2.statmt.org/wmt24/pdf/2024.wmt-1.1.pdf.

Lai, V., Nguyen, C., Ngo, N., Nguyen, T., Dernoncourt, F., Rossi, R., and Nguyen, T. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2023. URL https://aclanthology.org/2023.emnlp-demo.28.

Li, T., Chiang, W.-L., Frick, E., Dunlap, L., Wu, T., Zhu, B., Gonzalez, J. E., and Stoica, I. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*, 2024. URL https://arxiv.org/pdf/2406.11939.

Lin, X. V., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., Ott, M., Goyal, N., Bhosale, S., Du, J., Pasunuru, R., Shleifer, S., Koura, P. S., Chaudhary, V., O'Horo, B., Wang, J., Zettlemoyer, L., Kozareva, Z., Diab, M., Stoyanov, V., and Li, X. Few-shot learning with multilingual generative language models. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9019–9052, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.616. URL https://aclanthology.org/2022.emnlp-main.616/.

Liu, J., Xia, C. S., Wang, Y., and Zhang, L. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36, 2024.

Lu, Y., Zhu, W., Li, L., Qiao, Y., and Yuan, F. Llamax: Scaling linguistic horizons of llm by enhancing translation capabilities beyond 100 languages. *arXiv preprint arXiv:2407.05975*, 2024. URL https://arxiv.org/pdf/2407.05975.

Muennighoff, N., Wang, T., Sutawika, L., Roberts, A., Biderman, S., Le Scao, T., Bari, M. S., Shen, S., Yong, Z. X., Schoelkopf, H., Tang, X., Radev, D., Aji, A. F., Almubarak, K., Albanie, S., Alyafeai, Z., Webson, A., Raff, E., and Raffel, C. Crosslingual generalization through multitask finetuning. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15991–16111, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.891. URL https://aclanthology.org/2023.acl-long.891/.

OpenAI. Hello gpt-4o, 2024. URL https://openai.com/index/hello-gpt-4o/.

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L.,

Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Łukasz Kaiser, Kamali, A., Kanitscheider, I., Keskar, N. S., Khan, T., Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, J. H., Kiros, J., Knight, M., Kokotajlo, D., Łukasz Kondraciuk, Kondrich, A., Konstantinidis, A., Kosic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D., Mu, T., Murati, M., Murk, O., Mély, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Ouyang, L., O'Keefe, C., Pachocki, J., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Michael, Pokorny, Pokrass, M., Pong, V. H., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selsam, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J., Tezak, N., Thompson, M. B., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Twork, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W., and Zoph, B. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/

b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Ponti, E. M., Glavaš, G., Majewska, O., Liu, Q., Vulić, I., and Korhonen, A. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020. URL https://aclanthology.org/2020.emnlp-main.185.

Qi, J., Fernández, R., and Bisazza, A. Cross-lingual consistency of factual knowledge in multilingual language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://aclanthology.org/2023.emnlp-main.658.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Rein, D., Hou, B. L., Stickland, A. C., Petty, J., Pang, R. Y., Dirani, J., Michael, J., and Bowman, S. R. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, abs/2311.12022, 2023. URL https://arxiv.org/pdf/2311.12022.

Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023. URL https://arxiv.org/pdf/2308.12950.

Shi, F., Suzgun, M., Freitag, M., Wang, X., Srivats, S., Vosoughi, S., Chung, H. W., Tay, Y., Ruder, S., Zhou, D., Das, D., and Wei, J. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=fR3wGCk-IXp.

Singh, S., Romanou, A., Fourrier, C., Adelani, D. I., Ngui, J. G., Vila-Suero, D., Limkonchotiwat, P., Marchisio, K., Leong, W. Q., Susanto, Y., Ng, R., Longpre, S., Ko, W.-Y., Smith, M., Bosselut, A., Oh, A., Martins, A. F. T., Choshen, L., Ippolito, D., Ferrante, E., Fadaee, M., Ermis, B., and Hooker, S. Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation, 2024. URL https://arxiv.org/abs/2412.03304.

Snover, M. G., Dorr, B. J., Schwartz, R. M., Micciulla, L., and Makhoul, J. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers, AMTA 2006, Cambridge, Massachusetts, USA, August 8-12, 2006*, pp. 223–231.

Association for Machine Translation in the Americas, 2006. URL https://aclanthology.org/2006.amta-papers.25/.

Srinivasan, V. K., Dong, Z., Zhu, B., Yu, B., Mosk-Aoyama, D., Keutzer, K., Jiao, J., and Zhang, J. Nexusraven: a commercially-permissive language model for function calling. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023. URL https://openreview.net/pdf?id=5lcPe6DqfI.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. URL https://www.sciencedirect.com/science/article/pii/S0925231223011864.

Sun, Q., Chen, Z., Xu, F., Cheng, K., Ma, C., Yin, Z., Wang, J., Han, C., Zhu, R., Yuan, S., et al. A survey of neural code intelligence: Paradigms, advances and beyond. *arXiv preprint arXiv:2403.14734*, 2024. URL https://arxiv.org/pdf/2403.14734.

Talmor, A., Herzig, J., Lourie, N., and Berant, J. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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)*, 2019. URL https://aclanthology.org/N19-1421.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024. URL https://arxiv.org/abs/2408.00118.

Team, I. Internlm: A multilingual language model with progressively enhanced capabilities, 2023.

Wang, B., Liu, Z., Huang, X., Jiao, F., Ding, Y., Aw, A., and Chen, N. SeaEval for multilingual foundation models: From cross-lingual alignment to cultural reasoning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2024a. URL https://aclanthology.org/2024.naacl-long.22.

Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., and Sui, Z. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024b. URL https://aclanthology.org/2024.acl-long.511/.

Yuan, F., Shou, L., Bai, X., Gong, M., Liang, Y., Duan, N., Fu, Y., and Jiang, D. Enhancing answer boundary detection for multilingual machine reading comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.acl-main.87.

Zhang, Y., Deng, B., Wan, Y., Yang, B., Wei, H., Huang, F., Yu, B., Lin, J., Huang, F., and Zhou, J. P-mmeval: A parallel multilingual multitask benchmark for consistent evaluation of llms, 2024. URL https://arxiv.org/abs/2411.09116.

Zhou, J., Lu, T., Mishra, S., Brahma, S., Basu, S., Luan, Y., Zhou, D., and Hou, L. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023. URL https://arxiv.org/pdf/2311.07911.

Zhu, W., Liu, H., Dong, Q., Xu, J., Huang, S., Kong, L., Chen, J., and Li, L. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, 2024. URL https://aclanthology.org/2024.findings-naacl.176.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 3530–3534, 2016. URL https://conferences.unite.un.org/UNCorpus/Content/Doc/un.pdf.

A. Capability and Task Data Selection

Instruction Following Capability involves understanding and executing commands accurately and efficiently. In the light of varied evaluation methods - rule-based or model-based - we include two distinct tasks.

- Rule-based Intruction Following: We collect data from IFEval (Zhou et al., 2023), which is a benchmark for evaluating the instruction following abilities of LLMs, composed of around 500 verifiable instructions and can be evaluated for accuracy using automated rules. Note that the accuracy for IFEval is the average of the four accuracies (i.e. prompt-strict, prompt-loose, inst-strict and inst-loose accuracies), following (Dubey et al., 2024).
- Model-based Instruction Following: We collect data from Arena-hard (Li et al., 2024) which contains 500 real-world instructions from the Chatbot Arena (Chiang et al., 2024), and m-ArenaHard[6] which contains translated multilingual versions. This benchmark can provide better model separability and higher alignment with human preference. It is assessed by the Win Rate of the testing model in comparison to the baseline model, GPT-4o, judged against GPT-4o-mini.

Code Generation Capability refers to automatically producing functional code scripts based on given requirements. Considering variations in difficulty, two separate tasks are included.

- Function Completion: We collect data from Humaneval+ (Liu et al., 2024) which is an augmented version of HumanEval (Chen et al., 2021), comprising an expanded test cases. Each problem in the benchmark gives a definition of a Python function accompanied by an English docstring, and requires LLMs to complete the function.
- Problem Solving: We collect data from Live-CodeBench [7] (Jain et al., 2024) which provides a more rigorous assessment of the code generation capabilities. It is a much harder benchmark by collecting coding problems in natural language from real competition platforms with live updates.

Long Context Modeling Capability involves understanding and generating coherent text from extensive input sequences, allowing the model to capture dependencies and relationships within lengthy texts. This paper focuses on the long-context evaluation of multilingual settings based on the RULER benchmark (Hsieh et al., 2024).

---

[6]https://huggingface.co/datasets/CohereForAI/m-ArenaHard

[7]We adopt the code generation subset in LiveCodeBench v4 as the original English dataset.

- Question Answering: We build synthetic testsets based on RULER, which contains several question answering long-context tasks with pre-defined context length, such as the needle-in-a-haystack (NIAH) test and question answering (QA) test. Since the NIAH test is unrealistic and many models perform perfectly on it, we add a new task called QA-in-a-heystack (QAIAH), where one or several paragraphs are inserted into the haystack. The model then answers the question related to the inserted paragraph instead of finding the obtrusive needle. We reserve the tasks of NIAH, QAIAH, and variable tracking (VT) in our task list, while others are excluded.

Reasoning encompasses thinking logically, drawing conclusions, making inferences, and solving problems by processing data, applying rules, and utilizing various forms of logic and knowledge representation. Pushing LLMs beyond surface-level tasks, we extend MGSM (Shi et al., 2023) and GPQA (Rein et al., 2023) requiring deeper understanding and reasoning across different context.

- Math Reasoning: We collect data from MGSM which evaluates the capability of LLM to solve math reasoning problems in multiple languages, focusing on grade-school level complexity.
- Science Reasoning: We collect data from GPQA which is crucial for assessing LLM capability for advanced, unsearchable reasoning and critical thinking across diverse, complex domains. It comprises multiple choice questions formulated by experts in the domains of biology, physics, and chemistry, posing extreme challenges where human experts achieve accuracy lower than 70%.

Tool Use Capability requires the model to translate user queries into executable functions for calling in operating software tools. We extend Nexus (Srinivasan et al., 2023) to a multilingual version, which is adopted by Llama3 (Dubey et al., 2024).

- Multiple Functions: Nexus offers a set of functions and user queries. For each query, the language model is required to generate a function call from a list of noisy functions, in accordance with the function definitions and docstrings.

Translation Capability needs the model to convert text between multiple languages while maintaining semantic meaning accurately. To comprehensively evaluate this capability, we introduce general and task-specific translation datasets.

- General: General domain data are composed of Flores-200 (Costa-jussà et al., 2022), TED (Cettolo et al., 2012) and WMT24 (Kocmi et al., 2024) testsets. In BenchMAX, we include parallel data from 17 selected languages.
- Domain: Domain translation data is a by-product of the

Table 9. The recall of keywords when translating IFEval English data to other languages.

	zh	es	fr	de	hu	ru	ja	th	sw	bn	te	ar	ko	vi	cs	sr
w/o special symbols	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68	0.68
+symbol 1	0.91	0.89	0.88	0.92	0.93	0.93	0.91	0.91	0.92	0.95	0.96	0.90	0.88	0.90	0.92	0.99
+symbol 2	0.93	0.93	0.90	0.92	0.95	0.94	0.92	0.93	0.93	0.97	0.99	0.94	0.91	0.92	0.93	1.00

BenchMAX construction process, encompassing a 17-way parallel task across diverse domains, such as reasoning, code generation, tool usage, and instruction following. Unlike traditional translation tasks, this poses a new challenge to the model by requiring it to determine whether a given segment should be translated or not.

B. Dataset Construction

B.1. Rule-based Instruction Following Dataset

We first filter out some English-specific instructions from the original dataset, such as changing the letter cases. After filtering, the number of remaining samples is 429. The next problem is how to extract the keywords from the translated instruction since the keywords are also translated and are required in the verification step. We try different groups of special symbols to extract translated keywords. The recall rates are presented in Table 5, where order 1 achieves the best performance. Complete results across all languages are provided in Table 9. In addition, the number-word constraints for non-English languages are multiplied by a ratio in order to make the difficulty of the same instruction in different languages comparable. Specifically, we calculate the ratio of the word count of English to that of a non-English language in the Flores-200 corpus using language-specific tokenizers. we also adapt verification rules to multilingual scenarios. For instance, word and sentence segmentation methods may vary across different languages.

During post-editing, we ask human annotators to check whether the translated keywords in the kwargs, which are used by the rule-based program, appear in the translated instruction.

B.2. Model-based Instruction Following Dataset

Ten of the sixteen languages required have been provided by m-ArenaHard, which has translated the original dataset into 22 languages using Google Translate. Based on m-ArenaHard, we further translate the English data into six other languages via Google Translate. Subsequently, we ask human annotators to review and edit the translated instructions in all 16 languages.

B.3. Function Completion Dataset

The objective is to translate only the natural texts within the function comments. However, it is challenging to prevent Google Translate from translating other elements, such as function names. Alternatively, we instruct GPT-4o to complete this translation task with well-designed prompts (Table 16). Furthermore, a human post-editing process is employed to refine the quality of the generated translation.

B.4. Problem Solving Dataset

Similar to the Function Completion Dataset, we employ GPT-4o to translate the English problems into other 16 languages with a well-designed prompt (Table 17), since Google Translate cannot distinguish the parts that should remain untranslated. Human review is also used to ensure the overall quality of the translated texts.

B.5. Math Reasoning Dataset

Given that the MGSM examples are written in ten languages we need, we only translate the English version into the remaining six languages via Google Translate. This is also followed by a manual checking procedure.

B.6. Science Reasoning Dataset

The question and the four options of each sample are translated into 16 other languages by Google Translate. In particular, the question and options are concatenated by option markers like "(A)". After translation, we extract the translated question and options to form a new sample.

B.7. Long-Context Question Answering Dataset

The haystacks, needles, paragraphs and questions related to QAs are translated to other languages. We use the parallel testsets from the UN corpus (Ziemski et al., 2016) as the haystack. The English version contains about 128k tokens, and we extend it to other languages using Google Translate. The sentence of the needle is also translated into 16 other languages, in which UUIDs are employed as keys and values that are not translated. With respect to the QA data, we translate the paragraphs and questions in XQUAD (Artetxe et al., 2020) to the languages we need. Note that we also use the trick in translating IFEval to extract the answer
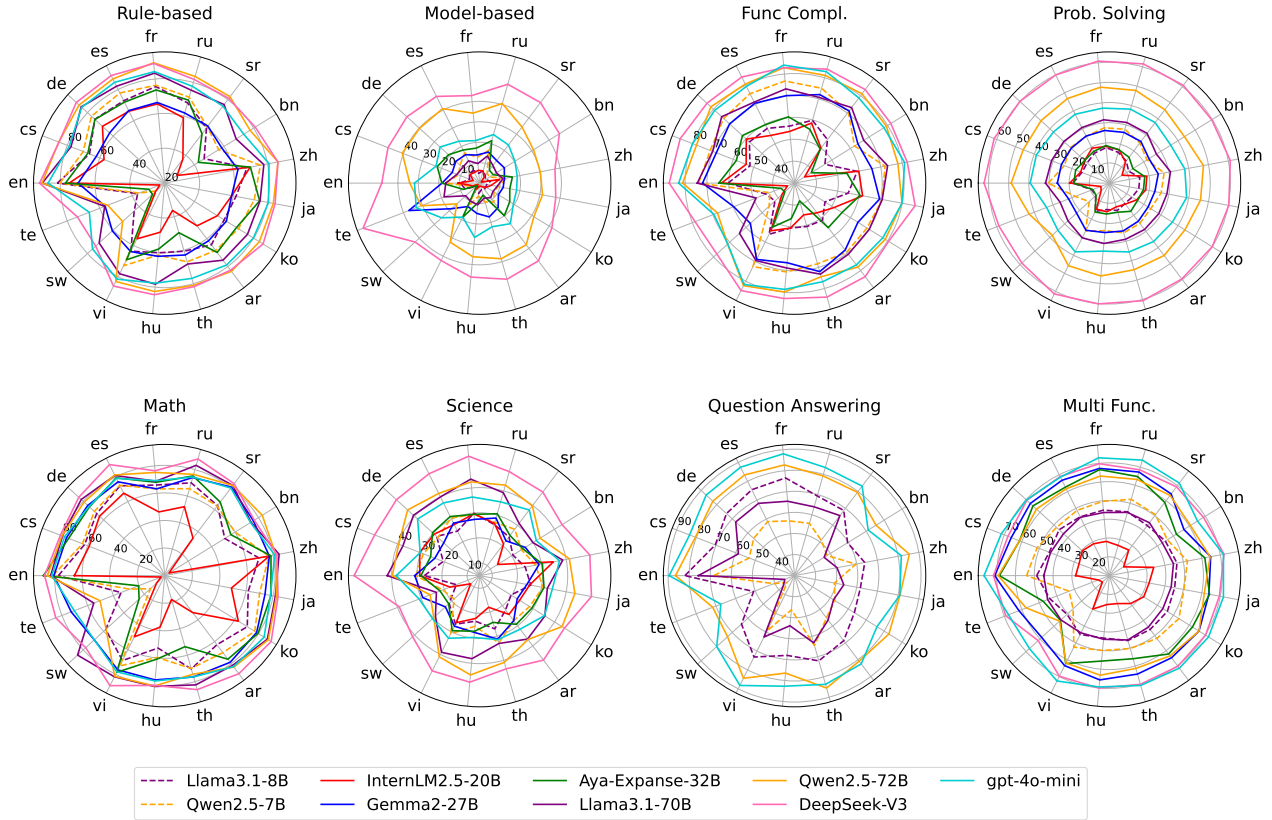
Figure 9. The radar charts visualize the performance of models on each subtask in different languages. Most model evaluated have imbalanced performance across different languages.

spans. With access to our multilingual haystacks, needles and paragraphs, we are able to synthesize the multilingual long-context testsets.

B.8. Multiple Functions Dataset

We only translate the user queries from English into other languages, given that the majority tool descriptions are written in English. The user queries are initially translated by Google Translate and subsequently adjusted by human annotators. To preserve the English parameters, we replace them with placeholders before machine translation and restore them afterward.

C. Model Information

Here we list the evaluated models in this section.

Llama3.1-Instruct (Dubey et al., 2024) series contains three multilingual large language models with number of parameters ranging from 8B to 405B. The pre-training corpus of Llama3.1 contains 8% multilingual tokens, and multilingual alignment is also optimized during post-training. In our experiments, we evaluate the 8B version and the 70B

version of Llama3.1-Instruct.

Qwen2.5-Instruct (Qwen Team, 2024) is a collection of multilingual language models with several sizes, ranging from 0.5B to 72B. The models are trained with multilingual tokens in both pre-training stage and post-training stage, and are rigorously evaluated on several multilingual tasks. In our experiments, we evaluate the 7B version and the 72B version of Qwen2.5-Instruct.

Aya-Expanse (Dang et al., 2024) is an open-weight research of models with advanced multilingual capabilities, supporting 23 languages. The Aya Expanse 8B and 32B variants are instruction-tuned and beat Llama3.1-instruct models on the m-ArenaHard, a multilingual instruction following benchmark.

Gemma2-IT (Team et al., 2024) family demonstrates strong multilingual capabilities, although this is not highlighted in the technical report. We benchmark the 9B and 27B variants of Gemma2-IT.

InternLM2.5-chat (Cai et al., 2024) is the successor of InternLM (Team, 2023), which is claimed as a multilingual model. We include the 7B version and 20B version in our experiments. InternLM2.5-7B-chat-1m is a long-context variant supporting context windows with 1M tokens.

DeepSeek-V3 (DeepSeek-AI et al., 2024) is one of state-of-the-art open-source models that achieve performance comparable to that of the best proprietary models. It is a 671B MoE model, with 37B activated for each token. A multilingual corpus and a multilingual-optimized tokenizer are incorporated into their training process.

GPT-4o & GPT-4o-mini (OpenAI, 2024) are two of the best proprietary models that also achieve remarkable performance on multilingual tasks. Their tokenizer can better compress multilingual texts than that of GPT-4. GPT-4o-mini is the smaller version of GPT-4o with powerful performance.

D. Details about Prompt Templates

We present the prompt templates used in each task in this section. Table 10 and Table 11 show the native-CoT prompts for MGSM and GPQA. Table 12 shows the prompt templates for some tasks where the original English template is used. Table 13 shows the prompt templates of the long-context modelling task. Table 15 shows the LLM-Judge

Instruction for comparing two translations.

Table 10. The native-CoT prompts of the mathematical reasoning task.

Language	Prompt
En	Question: {question}\nStep-by-Step Answer:
Zh	问题: {question}\n逐步解答:
Es	Pregunta: {question}\nRespuesta paso a paso:
Fr	Question : {question}\nRéponse étape par étape :
De	Frage: {question}\nSchritt-für-Schritt-Antwort:
Ru	Задача: {question}\nпошаговоерешение:
Ja	問題: {question}\nステップごとの答え:
Th	โจทย์: {question}\nคำตอบทีละขั้นตอน:
Sw	Swali: {question}\nJibu la Hatua kwa Hatua:
Bn	প্রশ্ন: {question}\nধাপে ধাপে উত্তর:
Te	ప్రశ్న: {question}\nదశలవారీగా సమాధానం:
Ar	السؤال: {question}\nالإجابة خطوة بخطوة:
Ko	질문: {question}\n단계별 답변:
Vi	Câu hỏi: {question}\nCâu trả lời từng bước:
Cs	Otázka: {question}\nOdpověď krok za krokem:
Hu	Kérdés: {question}\nVálasz lépésről lépésre:
Sr	Питање: {question}\nОдговор корак по корак:

E. Detailed results

Figure 9 illustrates the detailed results of each model on each task.

17

Table 11. The native-CoT prompts of the scientific reasoning task.

Language	Prompt
En	What is the correct answer to this question:{question}\nChoices:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nLet's think step by step:
Zh	这个问题的正确答案是什么: {question}\n选项:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\n我们来一步步思考一下:
Es	¿Cuál es la respuesta correcta a esta pregunta? {question}}\nOpciones:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nPensemos paso a paso:
Fr	Quelle est la bonne réponse à cette question : {question}\nChoix :\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nRéfléchissons étape par étape :
De	Was ist die richtige Antwort auf diese Frage: {question}\nAuswahlmöglichkeiten:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nLassen Sie uns Schritt für Schritt überlegen:
Ru	Какой правильный ответ на этот вопрос: {question}\nВарианты:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nДавайте подумаем шаг за шагом:
Ja	この質問の正しい答えは何ですか: {question}\n選択肢:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nステップごとに考えてみましょう:
Th	คำตอบที่ถูกต้องสำหรับคำถามนี้คืออะไร: {question}\nตัวเลือก:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nมาคิดทีละขั้นตอนกัน:
Sw	Je, ni jibu gani sahihi kwa swali hili: {question}\nChaguo: \n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nWacha tufikirie hatua kwa hatua:
Bn	এই প্রশ্নের সঠিক উত্তর কি: {question}\nপছন্দ:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nআসুন ধাপে ধাপে চিন্তা করি:
Te	ఈ ప్రశ్నకు సరైన సమాధానం ఏమిటి: {question}\nఎంపికలు:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nదశలవారీగా ఆలోచిద్దాం:
Ar	ما هي الإجابة الصحيحة لهذا السؤال: {question}\nالخيارات:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nلنفكر خطوة بخطوة:
Ko	이 질문에 대한 정답은 무엇입니까? {question}\n선택지:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\n단계별로 생각해 보겠습니다:
Vi	Câu trả lời đúng cho câu hỏi này là gì: {question}\nCác lựa chọn:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nChúng ta hãy suy nghĩ từng bước một:
Cs	Jaká je správná odpověď na tuto otázku: {question}\nMožnosti:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nZamysleme se krok za krokem:
Hu	Mi a helyes válasz erre a kérdésre: {question}\nVálasztási lehetőségek:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nGondoljuk végig lépésről lépésre:
Sr	Који је тачан одговор на ово питање: {question}\nИзбори:\n(A) {choice1}\n(B) {choice2}\n(C) {choice3}\n(D) {choice4}\nХајде да размислимо корак по корак:

Table 12. The prompt templates of the listed tasks. The prompt in the template is multilingual.

Task	Prompt Template
Rule-based instruction following	{prompt}
Model-based instruction following	{prompt}
Problem Solving	**[System Message]** You are an expert Python programmer. You will be given a question (problem specification) and will generate a correct Python program that matches the specification and passes all tests. You will NOT return anything except for the program. **[User Message]** ### Question: {question} ### Format: Read the inputs from stdin solve the problem and write the answer to stdout (do not directly test on the sample inputs). Enclose your code within delimiters as follows. ` ``python` # YOUR CODE HERE ` `` ` ### Answer: (use the provided format with backticks)
Function Completion	**[User Message]** Please provide a self-contained Python script that solves the following problem in a markdown code block: ` `` ` {prompt} ` `` `  **[Assistant Message]** Below is a Python script with a self-contained function that solves the problem and passes corresponding tests: ` ``python`
Tool use	**[Tool Info]** {prompt}

Table 13. The prompt templates of the long-context modelling task.

Subtask	Prompt Template
NIAH	[User Message] Some special magic uuids are hidden within the following text. Make sure to memorize it. I will quiz you about the uuids afterwards. {heystack} What are all the special magic uuids for {query} mentioned in the provided text? [Assistant Message] The special magic uuids for {query} mentioned in the provided text are
QA in a heystack (QAIAH)	[User Message] Answer the questions based on the given documents. Only give me the answers and do not output any other words. The following are given documents. {context} Answer the questions based on the given documents. Only give me the answers and do not output any other words. Questions: {query} [Assistant Message] Answers:
Variable Tracking (VT)	[User Message] Memorize and track the chain(s) of variable assignment hidden in the following text. {context} Question: Find all variables that are assigned the value {query} in the text above. [Assistant Message] Answer: According to the chain(s) of variable assignment in the text above, 5 variables are assgined the value {query}, they are:
QA	[User Message] Answer the question based on the given documents. Only give me the answer and do not output any other words. The following are given documents. {context} Answer the question based on the given documents. Only give me the answer and do not output any other words. Question: {query} [Assistant Message] Answer:

Table 14. The GEMBA-SQM prompt.

Score the following translation from {src_lang} to {tgt_lang} with respect to the human reference on a continuous scale from 0 to 100 that starts with "No meaning preserved", goes through "Some meaning preserved", then "Most meaning preserved and few grammar mistakes", up to "Perfect meaning and grammar"

{src_lang} source: "{source}"
{tgt_lang} translation: "{target}"
Score:

Table 15. LLM-Judge Instruction

[System Message]
Please act as an impartial judge and evaluate the quality of the lang translations provided by two humans for the English source sentence displayed below. You will be given human A's translation and human B's translation. Your job is to evaluate which human's translation is better.

You must identify and correct any mistakes or inaccurate information.

Consider if the human's translations are accurate and fluent. Accurate means the translation conveys the same meaning, information, and nuances as the original source text. Fluent refers to the quality of the translation in terms of its naturalness, readability, and adherence to the grammatical, stylistic, and idiomatic conventions of the target language.

Then consider whether the human's translations are consistent with the context. Code input/output and programming language syntax should not be translated. Finally, review the formatting of the translated text, including indentation, to ensure it is consistent and appropriate.

After providing your explanation, you must output only one of the following choices as your final verdict with a label:

1. Human A is significantly better: [[A>>B]]
2. Human A is slightly better: [[A>B]]
3. Tie, relatively the same: [[A=B]]
4. Human B is slightly better: [[B>A]]
5. Human B is significantly better: [[B>>A]]

Example output: "My final verdict is tie: [[A=B]]".
[User Message]
<| Source Text|>
{source}

<|The Start of Human A's Translation|>
{translation_1}
<|The End of Human A's Translation|>

<|The Start of Human B's Translation|>
{translation_2}
<|The End of Human B's Translation|>

Table 16. Prompt for translating the Function Completion task.

[System Message]
You are a professional translator specializing in technical content. Please translate the following English Python codes into {tgt_lang}, adhering to these specific guidelines:

1. **Do not translate** content representing code input/output or programming language syntax. Only translate content in comments.
2. **Maintain the original formatting** of the text, structure and indentation.
3. **Do not translate** any LaTeX code.
4. **Only output the translation** without any additional comments or explanations.
[User Message]
{problem}

Table 17. Prompt for translating the Problem Solving task.

[System Message]
You are a professional translator specializing in technical content. Please translate the following English coding problems into {tgt_lang}, adhering to these specific guidelines:

1. **Do not translate** any LaTeX code.
2. **Do not translate** content representing code input/output or programming language syntax.
3. **Maintain the original formatting** of the text and structure.
4. **Only output the translation** without any additional comments or explanations.
[User Message]
{problem}