MUG-Eval: A Proxy Evaluation Framework for Multilingual Generation Capabilities in Any Language

Seyoung Song[♡*] **Seogyeong Jeong**[♡*] **Eunsu Kim**[♡] **Jiho Jin**[♡] **Dongkwan Kim**[♡]
Jamin Shin[♣] **Alice Oh**[♡]

[♡]KAIST [♣]Trillion Labs

{seyoung.song, sg.jeong28, kes0317, jinjh0123, dongkwan.kim}@kaist.ac.kr
jay@trillionlabs.co, alice.oh@kaist.edu

Abstract

Evaluating text generation capabilities of large language models (LLMs) is challenging, particularly for low-resource languages where methods for direct assessment are scarce. We propose MUG-Eval, a novel framework that evaluates LLMs' multilingual generation capabilities by transforming existing benchmarks into conversational tasks and measuring the LLMs' accuracies on those tasks. We specifically designed these conversational tasks to require effective communication in the target language. Then, we simply use task success rate as a proxy for successful conversation generation. Our approach offers two key advantages: it is independent of language-specific NLP tools or annotated datasets, which are limited for most languages, and it does not rely on LLMs-as-judges, whose evaluation quality degrades outside a few high-resource languages. We evaluate 8 LLMs across 30 languages spanning high, mid, and low-resource categories, and we find that MUG-Eval correlates strongly with established benchmarks ($r > 0.75$) while enabling standardized comparisons across languages and models. Our framework provides a robust and resource-efficient solution for evaluating multilingual generation that can be extended to thousands of languages.

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities in many languages, but evaluating their multilingual generation abilities remains a significant challenge, particularly for low-resource languages. These challenges are particularly pronounced for low-resource languages, which often lack robust natural language processing tools, comprehensive reference corpora, or established benchmarks. Consequently, evaluation resources for these low-resource languages predominantly derive from massively multilingual evaluation benchmarks (Hasan et al., 2021; Goyal et al., 2022; Bandarkar et al., 2024; Adelani et al., 2024, *inter alia*). Extending and evaluating natural language generation tasks presents considerable complexity, especially in the absence of language-specific resources.

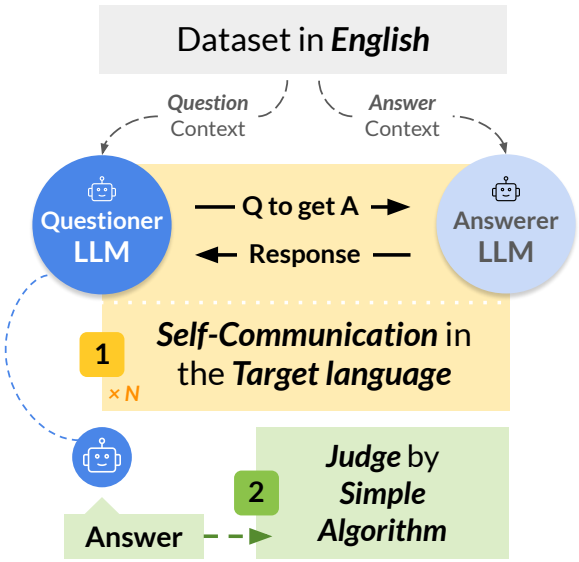

Figure 1: General concept of MUG-Eval. Two instances of the same LLM engage in self-communication in the target language to complete information-gap tasks. Model outputs are evaluated using algorithmic methods (*e.g.*, string matching or code testing), without requiring language-specific tools or LLMs-as-judges. Task success rate serves as a proxy for measuring the model's multilingual generation capability.

Recent approaches (Holtermann et al., 2024; Pombal et al., 2025) have employed LLMs-as-judges, but they face an inherent limitation—the reliability of judgments depends on the evaluator LLM's performance in the target language. While this limitation may be less pronounced for high-resource languages (Pombal et al., 2025), the applicability of such approaches to low-resource languages remains unclear and has not been rigorously validated. Conventional evaluation approaches for

[*]These authors contributed equally.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 19488–19514
November 4-9, 2025

Feature	Global-MMLU	Belebele	Flores-101	XL-Sum	MultiQ	MuG-Eval
Evaluates generation (not comprehension)	✗	✗	✓	✓	✓	✓
Metrics comparable across languages	✓	✓	✗	✗	✓	✓
No LLMs-as-Judges required	✓	✓	✓	✓	✗	✓
Native speaker annotation is optional	✗	✗	✗	✗	✓	✓
# of languages supported	42	122	101	47	137	2,102

Table 1: Positioning of MuG-Eval among multilingual evaluation benchmarks. MuG-Eval uniquely combines: (1) evaluation of generation capability (not just comprehension), (2) cross-linguistically comparable metrics, and (3) objective scoring without LLMs-as-judges, and (4) reduced dependency on cross-lingual annotation. Tested on 30 languages, MuG-Eval currently supports 2,102 languages via GlotLID (Kargaran et al., 2023), with the potential to scale further as more advanced language identification tools develop. Benchmarks referenced are MultiQ (Holtermann et al., 2024), Flores-101 (Goyal et al., 2022), XL-Sum (Hasan et al., 2021), Global-MMLU (Singh et al., 2025), and Belebele (Bandarkar et al., 2024).

generation ability often require human-annotated ground truth data, such as BLEU (Papineni et al., 2002) for machine translation or ROUGE (Lin, 2004) for summarization. Overall, there exists a gap in methodologies that offer both reliability and scalability for quantifying LLM generation performance across diverse languages.

In this paper, we propose MuG-Eval, a framework for evaluating the multilingual generation capabilities of LLMs, particularly for languages where direct evaluation proves challenging or infeasible. Our methodology creates information-gap scenarios that require successful communication in the target language to complete tasks, such as providing hidden information to one agent while another must discover it through questioning. We implement three tasks in MuG-Eval by adapting existing benchmarks into conversational and multilingual settings—Easy Twenty Questions (Zhang et al., 2024), MCQ Conversation (Bandarkar et al., 2024), and Code Reconstruction (Muennighoff et al., 2024)—where task completion rates serve as proxies for different aspects of generation ability: reasoning, instruction following, and programming (§3.1). Our approach builds on the insight from Muennighoff et al. (2024): instead of directly assessing LLM-generated text quality, we can indirectly measure how well the LLM comprehends what it has itself generated.

We evaluate 8 LLMs across 30 languages from high-, mid-, and low-resource categories as defined by Singh et al. (2024). Our experiments demonstrate that MuG-Eval has strong discriminative power, enabling precise comparisons both across languages and across models (§4.1). The framework shows high internal consistency among its three tasks and correlates strongly (Pearson's $r > 0.75$) with established benchmarks including Belebele (Bandarkar et al., 2024), MultiQ (Holtermann et al., 2024), and Global-MMLU (Singh et al., 2025) (§5.1). Additionally, our analysis of MCQ Conversation reveals that when native-language references are unavailable, English is not always the optimal substitute language, particularly for low-resource languages (§5.2).

Our primary contribution lies in proposing MuG-Eval[1], a novel language-agnostic framework for evaluating multilingual generation in large language models through self-comprehension tasks, without relying on language-specific NLP tools or human annotations. To demonstrate the utility and effectiveness of this framework, we structure the paper as follows. We begin by reviewing the landscape of multilingual generation evaluation, identifying critical gaps in existing methodologies that motivate our approach (§2). We then present the design of MuG-Eval, introducing three conversational tasks that recast generation evaluation as a communication-based task (§3). We evaluate eight large language models in 30 linguistically diverse languages, demonstrating strong correlations with established benchmarks while offering unprecedented scalability (§4). Through detailed analysis, we uncover cross-linguistic performance patterns and validate the effectiveness of MuG-Eval as a robust, language-agnostic evaluation framework (§5), and conclude with directions for future work in multilingual LLM evaluation (§6).

2 Related Work

Reference-based metrics such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and

[1]Code and dataset available at https://github.com/seyoungsong/mugeval.

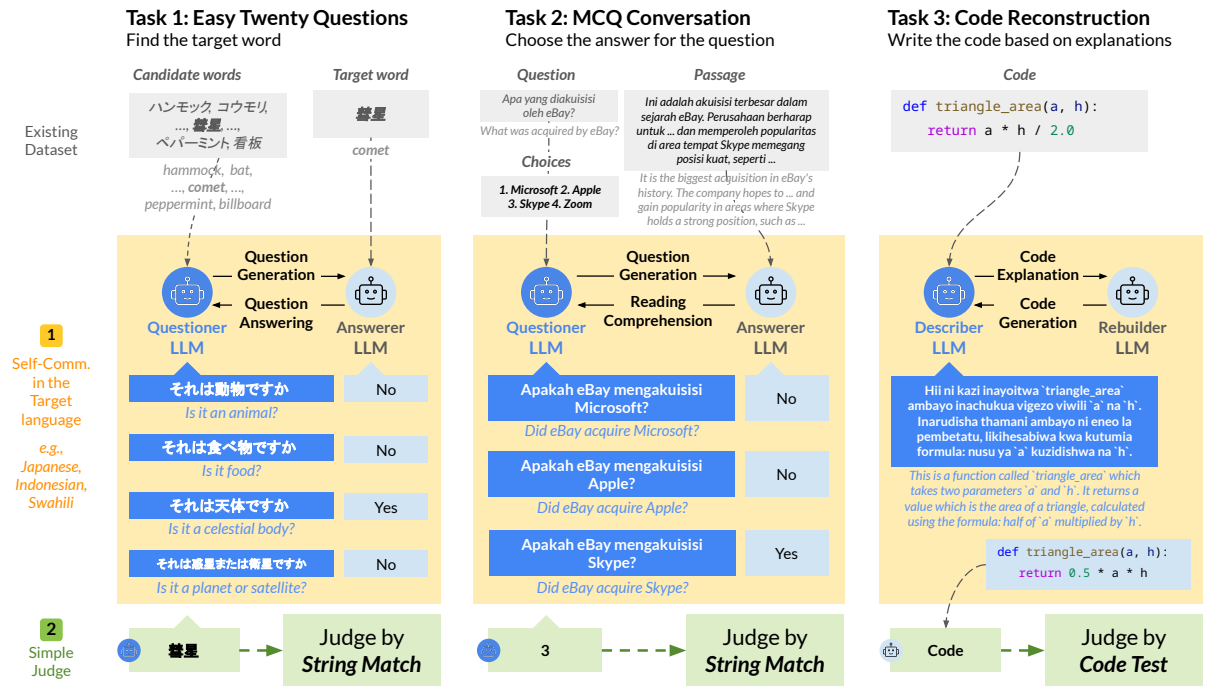

Figure 2: Overview of evaluation tasks. Two instances of the same LLM engage in self-communication in the target language to complete information-gap tasks: (1) Easy Twenty Questions—guessing a hidden word, (2) MCQ Conversation—finding the answer through passage-based dialogue, and (3) Code Reconstruction—explaining and reconstructing code.

chrF (Popović, 2015) assess generation quality by comparing outputs against reference texts, usually requiring human-generated target texts as ground truth. These metrics are widely adopted in benchmarks such as MEGA (Ahuja et al., 2023), GlotEval (Luo et al., 2025), Multi-IF (He et al., 2024), and BenchMAX (Huang et al., 2025). However, such reference-based approaches are limited by their reliance on high-quality parallel data, which is scarce in many languages. Moreover, they struggle in cross-lingual comparisons due to their sensitivity to lexical and syntactic features.

To address these limitations, reference-free methods—particularly those using LLMs as evaluators—gained attention (Dang et al., 2024; Holtermann et al., 2024; Pombal et al., 2025). Nonetheless, Hada et al. (2024b) highlights the instability and reduced reliability of LLM evaluators in low-resource or non-Latin script languages, raising concerns about fairness and generalizability.

An emerging line of work evaluates generation quality through downstream utility, assessing how well generated content supports task completion. Recent benchmarks explore the generation-comprehension link through interactive information-gap tasks that require mutual understanding. These include clarifying question generation (Gan et al., 2024), reference games (Gul and Artzi, 2024; Eisenstein et al., 2023), bidirectional code understanding (Muennighoff et al., 2024), and multi-turn interactive benchmarks such as HumanEvalComm (Wu and Fard, 2025), telephone-game simulations (Perez et al., 2025), and 20Q (Zhang et al., 2024).

Drawing inspiration from 20Q (Zhang et al., 2024) and HumanEvalExplain (Muennighoff et al., 2024), our framework builds on tasks that inherently require both comprehension and generation, foregrounding successful communication as the central evaluation criterion. Designed to be language-agnostic, reference-free, and LLM-independent, it offers a more equitable and scalable multilingual evaluation across an unlimited spectrum of languages.

3 MUG-Eval: A Language-Agnostic Evaluation Framework

MUG-Eval consists of three tasks adapted from existing benchmarks (Zhang et al., 2024; Bandarkar et al., 2024; Muennighoff et al., 2024) to evaluate multilingual generation capabilities. The benchmarks for Easy Twenty Questions and Code Reconstruction were originally English-only, while the

source for the MCQ Conversation task is the multilingual Belebele dataset. Each task is structured as a self-communication scenario between two "LLM instances"—separate API calls to the same model, each assigned a distinct conversational role (*e.g.*, Questioner or Answerer) with a unique system prompt and access to different information. The instances communicate turn-by-turn in the target language, with the output from one serving as the input for the next. The model's capability is measured by the task completion rate, which serves as the primary evaluation metric.

This section provides detailed descriptions of each task and evaluation procedures. Additional details, including prompts and generation parameters, are provided in the Appendix B.2.

3.1 Tasks

Easy Twenty Questions. This task evaluates reasoning and strategic questioning abilities through a word-guessing game. Drawing from the *Things* dataset (Zhang et al., 2024), we translate 140 English words into 30 languages using Google Translate. One model instance (answerer) receives a hidden word from this set, while another (questioner) must identify it from a list of 100 candidates. The questioner poses up to 20 yes/no questions in the target language, to which the answerer responds only with "yes," "no," or "maybe" in English. The predefined candidates ensure consistent evaluation across languages, mitigating lexical diversity from affecting task difficulty or scoring mechanisms.

MCQ Conversation. We transform the Belebele benchmark (Bandarkar et al., 2024)—a reading comprehension dataset spanning 122 languages—into a conversational task. From the original dataset of 900 samples, we separate the reading passages from their corresponding questions and answer choices. Similar to the previous task, the answerer instance accesses only the passage, while the questioner sees the question and four answer options. To discover the correct answer, the questioner may ask up to 10 yes/no questions in the target language, receiving "yes," "no," or "maybe" responses in English, similar to the previous task. This design tests multi-turn instruction-following capabilities.

Code Reconstruction. This task adapts HumanEvalExplain (Muennighoff et al., 2024) to assess code generation abilities across languages, not only in English. Using 164 Python function samples with corresponding unit tests, one model instance (describer) generates a natural language explanation of the code in the target language. Another instance (rebuilder) then reconstructs the original function from this description and the function declaration snippet. Success is measured by whether the reconstructed code passes all unit tests.

3.2 Evaluation Metrics

Task completion rate serves as our primary metric, calculated as the ratio of successfully completed tasks. We use exact string matching for word or choice predictions, with responses prompted to appear within double brackets and extracted via regular expressions. We employ GlotLID (Kargaran et al., 2023) to ensure the model's responses are in the target language. Tasks fail when models: (1) produce a question or description in the wrong language, (2) produce invalid responses, or (3) violate task-specific constraints such as including more than 20 consecutive source code characters in explanations.

4 Experiments

Models. We evaluate eight multilingual large language models to assess their generation capabilities across diverse languages. Our selection includes four open-weight models: Llama 3.3-70B (Llama Team, 2024), Llama 3.1-8B, Qwen2.5-72B (Qwen Team, 2024), and Qwen2.5-7B, alongside four closed-source models: GPT-4o (OpenAI, 2024), GPT-4o-mini, Gemini 2.5 Flash (Google, 2025), and Gemini 2.0 Flash (Google, 2024). All models are accessed via API endpoints, with GPT-4o variants served through Azure OpenAI Services and the remaining models through OpenRouter. Detailed model information is provided in the Appendix B.1.

Languages. We test our framework on 30 languages grouped by resource availability following Singh et al. (2024)'s classification, with 10 languages selected from each resource category. We include high-resource languages Arabic (`arb`), Chinese (`zho`), English (`eng`), French (`fra`), German (`deu`), Hindi (`hin`), Italian (`ita`), Japanese (`jpn`), Portuguese (`por`), and Spanish (`spa`); mid-resource languages Bengali (`ben`), Greek (`ell`), Hebrew (`heb`), Indonesian (`ind`), Korean (`kor`), Lithuanian (`lit`), Malay (`zsm`), Romanian (`ron`), Thai (`tha`), and Ukrainian (`ukr`); and low-resource languages Amharic (`amh`), Hausa (`hau`), Igbo (`ibo`),

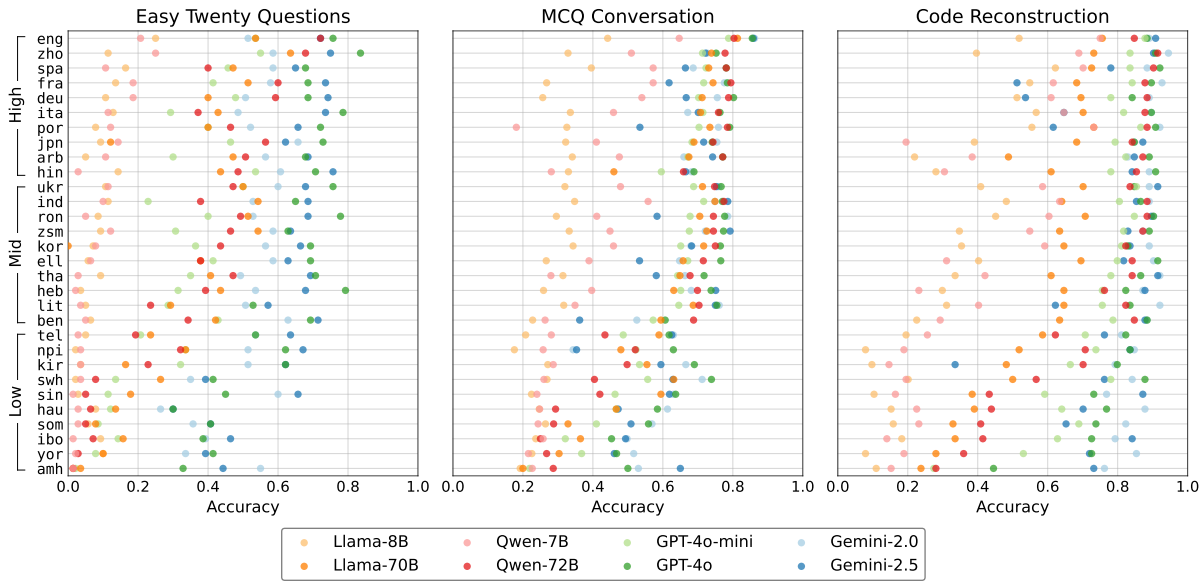

Figure 3: Accuracy of 8 LLMs across three tasks in 30 languages. Languages are grouped by resource level and sorted by average performance within each group. Results show that Code Reconstruction is the easiest task, followed by MCQ Conversation and Easy Twenty Questions. The gap is minor between high and mid-resource languages, but substantial between mid and low. Larger models consistently outperform smaller ones within the same language family, and tasks exhibit distinct ceiling effect.

Model	Easy Twenty Questions					MCQ Conversation					Code Reconstruction				
	All	ENG	High	Mid	Low	All	ENG	High	Mid	Low	All	ENG	High	Mid	Low
GPT-4o	62.21	**75.71**	**72.64**	**69.21**	44.79	**70.14**	85.56	**77.31**	**74.33**	**58.78**	83.43	88.41	89.02	86.59	74.70
Gemini-2.0-flash	51.93	51.43	56.07	55.57	44.14	66.72	**86.22**	73.33	69.74	57.08	**86.79**	89.02	**89.21**	**89.45**	**81.71**
Gemini-2.5-flash	**62.26**	72.14	70.57	66.36	**49.86**	62.90	85.89	68.90	65.74	54.07	77.05	**90.85**	74.63	84.39	72.13
Qwen2.5-72B	35.17	72.14	53.86	40.64	11.00	61.90	80.33	76.61	72.44	36.63	73.68	84.76	87.56	84.15	49.33
GPT-4o-mini	31.95	53.57	44.29	35.93	15.64	59.83	78.78	70.11	65.91	43.48	75.02	87.80	82.50	80.12	62.44
Llama-3.3-70B	33.79	53.57	44.14	40.36	16.86	61.15	81.33	70.04	68.29	45.12	58.03	75.61	68.05	65.61	40.43
Qwen2.5-7B	7.90	20.71	14.50	6.64	2.57	37.33	64.67	46.48	40.33	25.17	40.47	75.00	56.28	46.22	18.90
Llama-3.1-8B	8.45	25.00	12.64	7.71	5.00	28.94	44.22	33.46	30.23	23.13	31.95	51.83	46.10	36.16	13.60

Table 2: Average accuracy (%) of 8 LLMs across three tasks, grouped by language resource categories. The best and the second-best performances within each task and resource category are **bolded** and underlined, respectively. A consistent performance degradation is observed as the language resource level decreases from high (including English) to low.

Kyrgyz (kir), Nepali (npi), Sinhala (sin), Somali (som), Swahili (swh), Telugu (tel), and Yoruba (yor). This selection covers diverse language families and writing systems, including Latin, Cyrillic, and Devanagari scripts, ensuring comprehensive evaluation across typologically distinct languages. Detailed language information is provided in the Appendix A.1.

4.1 Results

Table 2 summarizes overall accuracy, and Figure 3 visualizes trends by language and task. Full results are provided in Appendix C.1.

How difficult is MUG-Eval? Average accuracy scores across tasks vary depending on the model and the resource level of the language. Code Reconstruction is the easiest task, followed by MCQ Conversation, while Easy Twenty Questions challenges the most. This may be due to the number of interaction turns: multi-turn tasks are more error-prone as mistakes accumulate. This pattern aligns with average turn counts (Table 9): Easy Twenty Questions requires the most turns, MCQ Conversation fewer, and Code Reconstruction only one.

Performance varies across resource levels and models. The performance gap between high- and mid-resource language groups is relatively small compared to the much larger gap observed between mid- and low-resource groups. Additionally, larger models consistently outperform smaller ones

within the same model family. Despite some variation in task-wise rankings, overall trends of task rankings remain stable across models.

Complementary ceiling effects exist across tasks. Code Reconstruction and MCQ Conversation saturate near the upper bound—around 0.9 and 0.8, indicating 90% and 80% accuracy. In contrast, Easy Twenty Questions exhibits saturation toward the lower end, with many scores concentrated near zero—especially in low-resource languages and smaller models. MCQ Conversation shows lower saturation than its original benchmark, Belebele (0.8 vs. 0.95; see Figure 4), likely due to its split-agent design, which can produce ambiguous question generations, leading to unsolvable cases.

These differing saturation patterns enhance the discriminative power of MUG-Eval. Easier tasks are more effective at separating weaker models and low-resource languages, while the harder task better distinguishes stronger models and high-resource languages. Together, they ensure that MUG-Eval maintains discriminative power across the full performance spectrum.

5 Discussion

5.1 Comparative Analysis

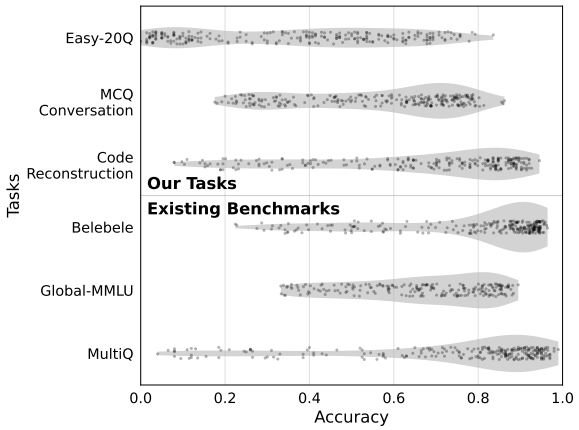

Figure 4: Score distributions across six evaluation tasks, demonstrating varying discriminative powers. Notably, MCQ Conversation, derived from the Belebele task, exhibits greater statistical dispersion, indicating greater ability to distinguish between models than the original Belebele benchmark.

Which tasks best distinguish between models? Figure 4 presents violin plots of accuracy scores for six tasks, including the three introduced in MUG-Eval. Easy Twenty Questions exhibited a broad distribution of scores, indicating strong discriminative power and the ability to distinguish models with varying capabilities. In contrast, Code Reconstruction showed a much narrower range, suggesting limited differentiation among a few models. Notably, MUG-Eval's MCQ Conversation demonstrated substantially greater discriminative power compared to the original Belebele task, highlighting its usefulness in evaluating multilingual understanding with finer granularity. Overall, all three tasks in MUG-Eval show greater discriminative capability than the three existing benchmarks.

How consistent is performance across different tasks? To validate the internal consistency of our framework, we analyzed performance correlations across our three tasks. While the tasks measure distinct abilities, a moderate positive correlation suggests that they capture a consistent, general signal of a model's multilingual capabilities. Figure 5 compares these performance correlations across six tasks, including the three introduced in MUG-Eval. Pearson correlation coefficients are all above 0.75, indicating strong consistency between task accuracy. Spearman's rank correlation coefficients exceed 0.75 in all cases, suggesting positive correlations in rank ordering. The reason why the correlations are not perfect is likely due to the distinct capabilities each task targets. Easy Twenty Questions primarily evaluates the reasoning aspect of generation, MCQ Conversation focuses on instruction following, Code Reconstruction assesses coding under information asymmetry. These differences account for the variation observed across tasks despite overall similarity.

Validation against established benchmarks. Figure 5 also compares performance correlations across six tasks, including the three introduced in MUG-Eval. While neither Pearson's nor Spearman's coefficients indicate perfect alignment between the three tasks in MUG-Eval and existing benchmarks, the figure demonstrates a high degree of correlation. This suggests that MUG-Eval produces reliable results in terms of both accuracy and ranking, despite its low cost due to the absence of human-annotated datasets. The detailed visualization result on Pearson's r is provided in Appendix C.2.

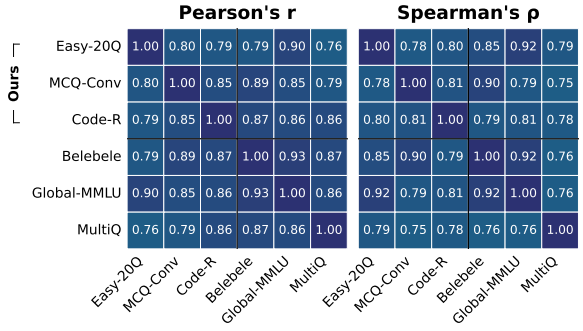

Figure 5: Correlation analysis between MUG-Eval tasks and existing multilingual benchmarks. Heatmaps show Pearson's r (left) and Spearman's ρ (right) correlation coefficients between three MUG-Eval tasks and three established benchmarks. All correlations exceed 0.75, demonstrating strong consistency between MUG-Eval and existing evaluation methods, validating its effectiveness as a multilingual evaluation framework.

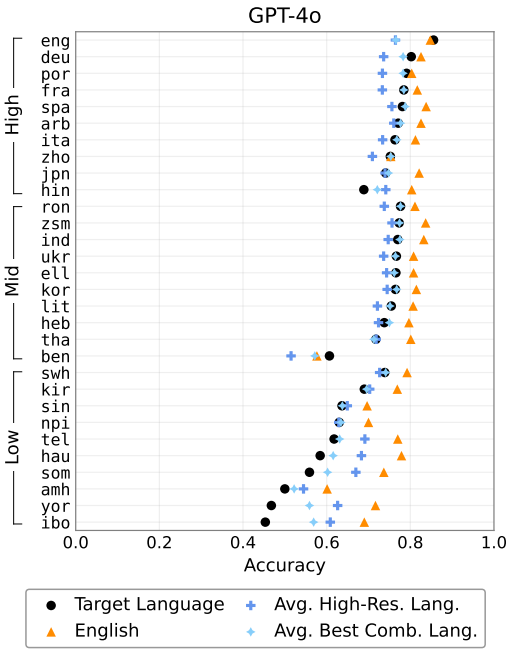

Figure 6: MCQ Conversation accuracy comparison across 30 languages for GPT-4o using passages in: (1) the target language, (2) English, and (3) five fixed high-resource languages (averaged), and (4)an optimized subset of up to five high-resource languages most similar to the target language. Results demonstrate that high-resource language substitution more closely approximates native language performance than using English alone, especially for low-resource languages.

5.2 Language Resource Flexibility: A Substitution Analysis

The original MCQ Conversation task assumes that the answerer receives a passage written in the target language. This raises a practical question: if such a passage is unavailable, can an English passage be used instead without significantly affecting performance? Would using passages from other high-resource languages yield a better substitute?

To investigate this, three experimental settings were compared: (1) using the original target language passage, (2) using an English passage, and (3) using five separate versions of each passage, each written in one of the high-resource languages—English, Chinese, Arabic, Japanese, or Hindi. Two models, GPT-4o and GPT-4o-mini, were evaluated,[1] with the GPT-4o result presented in Figure 6. The result on the other model (GPT-4o-mini) is provided in the Appendix C.3.

On average, performance based on the five high-resource language passages more closely approximated that of the target-language baseline than when using English alone. This indicates that incorporating diverse high-resource languages may provide a better alternative when native-language passages are unavailable.

To further validate the applicability of MCQ Conversation, we conducted an evaluation to assess whether replacing native-language passages with those in five high-resource languages maintains consistent performance patterns across languages. The correlation between results using original target-language passages and those using the high-resource substitutes was 0.60 for Pearson (based on raw scores) and 0.71 for Spearman (based on rank-order consistency). Given that MUG-Eval is ultimately designed for cross-lingual comparisons, the higher Spearman correlation suggests that relative language rankings are preserved without native-language input.

To deepen the analysis, we identified the high-resource language combination that best approximates the native passage for each target language. MCQ Conversation was executed across all target languages using the five high-resource passages across two models: GPT-4o and GPT-4o-mini.

For each case, the L2 distance between the performance with the substituted passage and that on the original native-language passage was calculated. The combination of high-resource language that minimizes this distance is reported in Table 7 and plotted in Figure 6. Results show that for high- and mid-resource languages, the best-performing

[1]This resource-intensive analysis was limited to the GPT models available via Azure OpenAI Service to stay within our computational budget.

combination typically includes English. However, for low-resource languages, combinations excluding English usually performed better. This indicates that English is not always the optimal substitute, especially for low-resource languages. The details about the best combinations on each language is provided in Appendix C.4.

5.3 Qualitative Error Analysis: GPT-4o in English and Korean

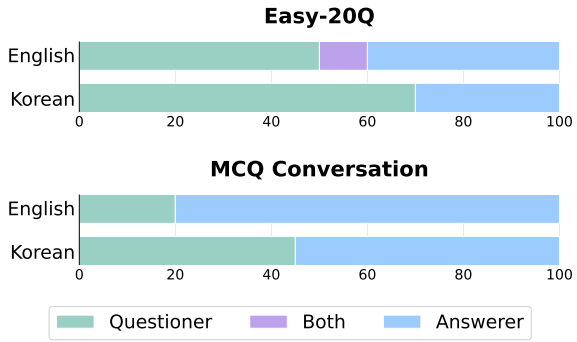

Figure 7: Attribution of errors by conversational role. Bars show the percentage of failures caused by Questioner (green), Answerer (blue), or Both roles (purple).

Setup. To validate that task completion rates reflect genuine language capabilities, we conducted a fine-grained error analysis on GPT-4o outputs in English and Korean. We chose GPT-4o as a representative high-performing model and selected English and Korean to leverage the authors' proficiency for reliable annotation. The authors manually annotated 160 GPT-4o conversation logs, sampling 20 success and failure cases each for Easy Twenty Questions and MCQ Conversation in English and Korean. Initial classification was performed using Gemini-2.5-flash, then manually corrected by two authors proficient in both languages.

Findings. Figure 7 reveals systematic task-specific error patterns that validate our framework design. The Code Reconstruction task is excluded from this role-based error analysis, as attributing failure to either the 'describer' or 'rebuilder' is inherently ambiguous. Easy Twenty Questions failed primarily due to questioner errors, reflecting strategic question generation challenges, while MCQ Conversation showed predominantly answerer errors, indicating passage comprehension difficulties. These patterns remained consistent across languages, confirming that failures stem from genuine communicative challenges rather than external factors. Success cases showed minimal errors in both roles, while rare successful cases with conversational errors reflected expected random chance. The LLM-based initial annotation achieved 78.8% accuracy (62.5% for failure cases, 95.0% for success cases).

Representative Error Case. In the MCQ Conversation task, Questioner errors often stemmed from failures to faithfully incorporate all relevant information from the original query when generating questions. Key semantic or lexical elements were frequently omitted, resulting in questions that lacked sufficient grounding in the passage—ultimately leading to unanswerable or misleading queries. In contrast, Answerer errors primarily reflected incorrect inference from the passage. Detailed examples of representative error cases are provided in Appendix C.5.

In the Easy Twenty Questions task, Questioner errors were typically caused by ineffective information-seeking strategies, such as asking insufficiently discriminative questions within the 20-turn limit or making premature guesses despite the presence of multiple plausible candidates. Most Answerer errors in this task were due to hallucinated responses, where the model generated logically incorrect "yes"/"no"/"maybe" answers.

5.4 Generation Statistics

While running the experiments, we collected detailed generation statistics, averaged over models and language groups. Specifically, we measured (1) token count, (2) sequence length, (3) language fidelity, (4) instruction-following of the Answerer, and (5) interaction length. A full description of these statistics is provided in Appendix D. We summarize key findings below:

- **Token Count and Sequence Length**: Output length varied by language resource level, with English being the shortest and low-resource languages generally producing the longest outputs.
- **Language Fidelity**: Although slightly lower in low-resource languages, fidelity scores remained similarly high across all groups.
- **Answerer Instruction-Following and Interaction Length**: These metrics were largely consistent across language resource groups

and models. On average, Easy Twenty Questions involved 14.3 turns, and MCQ Conversation 4.0.

6 Conclusion

A fundamental limitation in multilingual evaluation is the reliance on ground-truth references or LLM-based judgments, which are often unreliable or infeasible for low-resource languages. To address this, we introduce 🍺 **MUG-Eval**, a language-agnostic evaluation framework based on three conversational task completion between LLMs that assess both generation and comprehension.

We evaluate 8 LLMs across 30 languages using MUG-Eval. Our framework demonstrates strong internal consistency and aligns well with established multilingual benchmarks, while remaining reference-free and cost-effective. Our results highlight a few implications. First, MUG-Eval enables fine-grained performance comparisons even in low-resource settings due to its task diversity and saturation characteristics. Second, we find that substituting native-language passages with English often degrades performance—especially for low-resource languages—underscoring the need for evaluation methods that go beyond English-centric assumptions.

Limitations

MUG-Eval measures whether communication succeeds, but not how well it succeeds—a model generating minimal functional text scores identically to one producing sophisticated, nuanced output, as long as both complete the task. This limitation poses challenges for applications requiring natural, culturally appropriate, or stylistically rich text generation. Furthermore, comparing linguistic quality across languages remains fundamentally difficult because notions of richness and quality vary significantly across linguistic and cultural contexts, making it challenging to establish universal cross-linguistic metrics. This focus on communicative effectiveness over stylistic quality is an intentional design choice, ensuring our framework remains scalable and objective in low-resource settings where fluency evaluation is often infeasible. While this trade-off enables our language-agnostic evaluation approach, it remains a limitation for comprehensively assessing generation quality.

While MUG-Eval's reliability is supported by its strong correlations with existing benchmarks, comprehensive human evaluation has not yet been conducted. Our qualitative error analysis of 160 conversation logs (§5.3) provided initial validation of failure patterns and confirmed that task failures stem from genuine communicative challenges rather than external factors. However, broader human validation across all 30 languages would provide deeper insights into the framework's fairness across different languages and enable more detailed qualitative analysis of model performance patterns. Given the conversational nature of MUG-Eval's tasks, human evaluation could reveal which specific conversational aspects challenge different models, particularly since performance varies significantly depending on conversational roles.

Despite MUG-Eval's language-agnostic design, certain implementation aspects remain English-centric. The difficulty of accurately translating prompts into all target languages, especially low-resource ones, necessitated using English for instructional prompts in the conversational scenarios. Additionally, the Code Reconstruction task employs Latin script for code, with variable and function names following English naming conventions. These factors may introduce systematic biases against non-Latin script languages and low-resource language contexts, potentially affecting the framework's cross-linguistic validity.

Ethical Considerations

Our human evaluation study was conducted with approval from the Institutional Review Board (IRB), ensuring all procedures adhered to established ethical research standards. All participants recruited for the annotation task were compensated for their time at a rate of 30,000 KRW (approximately 21.76 USD as of September 2025), a rate that meets or exceeds fair compensation guidelines for our region.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00509258 and No. RS-2024-00469482, Global AI Frontier Lab).

This research project has benefitted from the Microsoft Accelerate Foundation Models Research (AFMR) grant program through which leading foundation models hosted by Microsoft Azure along with access to Azure credits were provided

to conduct the research.

We acknowledge using ChatGPT[1] and Claude[2] for writing and coding assistance, and Perplexity[3] and OpenScholar (Asai et al., 2024) for literature search.

References

David Ifeoluwa Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba O. Alabi, Yanke Mao, Haonan Gao, and En-Shiun Annie Lee. 2024. SIB-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 226–245, St. Julian's, Malta. Association for Computational Linguistics.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, David Wadden, Matt Latzke, Minyang Tian, Pan Ji, Shengyan Liu, Hao Tong, Bohao Wu, Yanyu Xiong, Luke Zettlemoyer, and 6 others. 2024. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *ArXiv preprint*, abs/2411.14199.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, and 26 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *ArXiv preprint*, abs/2412.04261.

Jacob Eisenstein, Vinodkumar Prabhakaran, Clara Rivera, Dorottya Demszky, and Devyani Sharma. 2023. MD3: the multi-dialect dataset of dialogues. In *24th Annual Conference of the International Speech Communication Association, Interspeech 2023, Dublin, Ireland, August 20-24, 2023*, pages 4059–4063. ISCA.

Yujian Gan, Changling Li, Jinxia Xie, Luou Wen, Matthew Purver, and Massimo Poesio. 2024. Clarq-llm: A benchmark for models clarifying and requesting information in task-oriented dialog. *ArXiv preprint*, abs/2409.06097.

Google. 2024. Introducing gemini 2.0: our new ai model for the agentic era.

Google. 2025. Gemini 2.5: Our most intelligent ai model.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Mustafa Omer Gul and Yoav Artzi. 2024. CoGen: Learning from feedback with coupled comprehension and generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12966–12982, Miami, Florida, USA. Association for Computational Linguistics.

Rishav Hada, Varun Gumma, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2024a. METAL: Towards multilingual meta-evaluation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2280–2298, Mexico City, Mexico. Association for Computational Linguistics.

Rishav Hada, Varun Gumma, Adrian de Wynter, Harshita Diddee, Mohamed Ahmed, Monojit Choudhury, Kalika Bali, and Sunayana Sitaram. 2024b. Are large language model-based evaluators the solution to scaling up multilingual evaluation? In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1051–1070, St. Julian's, Malta. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online. Association for Computational Linguistics.

Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, Shruti Bhosale, Chenguang Zhu, Karthik Abinav Sankararaman, Eryk Helenowski, Melanie Kambadur, Aditya Tayade, Hao Ma, Han

[1] https://chatgpt.com

[2] https://claude.ai

[3] https://perplexity.ai

Fang, and Sinong Wang. 2024. Multi-if: Benchmarking llms on multi-turn and multilingual instructions following. *ArXiv preprint*, abs/2410.15553.

Carolin Holtermann, Paul Röttger, Timm Dill, and Anne Lauscher. 2024. Evaluating the elementary multilingual capabilities of large language models with MultiQ. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4476–4494, Bangkok, Thailand. Association for Computational Linguistics.

Xu Huang, Wenhao Zhu, Hanxu Hu, Conghui He, Lei Li, Shujian Huang, and Fei Yuan. 2025. Benchmax: A comprehensive multilingual evaluation suite for large language models. *ArXiv preprint*, abs/2502.07346.

Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. 2023. GlotLID: Language identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore. Association for Computational Linguistics.

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

Llama Team. 2024. The llama 3 herd of models. *ArXiv preprint*, abs/2407.21783.

Hengyu Luo, Zihao Li, Joseph Attieh, Sawal Devkota, Ona de Gibert, Shaoxiong Ji, Peiqin Lin, Bhavani Sai Praneeth Varma Mantina, Ananda Sreenidhi, Raúl Vázquez, Mengjie Wang, Samea Yusofi, and Jörg Tiedemann. 2025. Gloteval: A test suite for massively multilingual evaluation of large language models. *ArXiv preprint*, abs/2504.04155.

Niklas Muennighoff, Qian Liu, Armel Randy Zebaze, Qinkai Zheng, Binyuan Hui, Terry Yue Zhuo, Swayam Singh, Xiangru Tang, Leandro von Werra, and Shayne Longpre. 2024. Octopack: Instruction tuning code large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

OpenAI. 2024. Gpt-4o contributions.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jérémy Perez, Grgur Kovač, Corentin Léger, Cédric Colas, Gaia Molinaro, Maxime Derex, Pierre-Yves Oudeyer, and Clément Moulin-Frier. 2025. When LLMs play the telephone game: Cultural attractors as conceptual tools to evaluate LLMs in multi-turn settings. In *The Thirteenth International Conference on Learning Representations*.

José Pombal, Dongkeun Yoon, Patrick Fernandes, Ian Wu, Seungone Kim, Ricardo Rei, Graham Neubig, and Andre Martins. 2025. M-prometheus: A suite of open multilingual LLM judges. In *Second Conference on Language Modeling*.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Qwen Team. 2024. Qwen2.5 technical report. *ArXiv preprint*, abs/2412.15115.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David Ifeoluwa Adelani, Jian Gang Ngui, Daniel Vila-Suero, Peerat Limkonchotiwat, Kelly Marchisio, Wei Qi Leong, Yosephine Susanto, Raymond Ng, Shayne Longpre, Sebastian Ruder, Wei-Yin Ko, Antoine Bosselut, Alice Oh, Andre Martins, Leshem Choshen, Daphne Ippolito, and 4 others. 2025. Global MMLU: Understanding and addressing cultural and linguistic biases in multilingual evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18761–18799, Vienna, Austria. Association for Computational Linguistics.

Shivalika Singh, Freddie Vargus, Daniel D'souza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O'Mahony, Mike Zhang, Ramith Hettiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, and 14 others. 2024. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567, Bangkok, Thailand. Association for Computational Linguistics.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Jie Jw Wu and Fatemeh H. Fard. 2025. Humanevalcomm: Benchmarking the communication competence of code generation for llms and llm agent. *ACM Trans. Softw. Eng. Methodol.* Just Accepted.

Yizhe Zhang, Jiarui Lu, and Navdeep Jaitly. 2024. Probing the multi-turn planning capabilities of LLMs via 20 question games. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1495–1516, Bangkok, Thailand. Association for Computational Linguistics.

Appendix

A Data Preparation

A.1 Languages

Throughout this paper, we evaluated LLMs across 30 languages: 10 high-resource, 10 mid-resource, and 10 low-resource languages. The resource classification follows the categorization defined by Singh et al. (2024).

ISO Code	Language	Script	Resources
arb_Arab	Arabic	Arabic	High
deu_Latn	German	Latin	High
eng_Latn	English	Latin	High
fra_Latn	French	Latin	High
hin_Deva	Hindi	Devanagari	High
ita_Latn	Italian	Latin	High
jpn_Jpan	Japanese	Japanese	High
por_Latn	Portuguese	Latin	High
spa_Latn	Spanish	Latin	High
zho_Hans	Chinese	Simplified Han	High
ben_Beng	Bengali	Bengali	Mid
ell_Grek	Greek	Greek	Mid
heb_Hebr	Hebrew	Hebrew	Mid
ind_Latn	Indonesian	Latin	Mid
kor_Hang	Korean	Hangul	Mid
lit_Latn	Lithuanian	Latin	Mid
ron_Latn	Romanian	Latin	Mid
tha_Thai	Thai	Thai	Mid
ukr_Cyrl	Ukrainian	Cyrillic	Mid
zsm_Latn	Malay	Latin	Mid
amh_Ethi	Amharic	Ethiopic	Low
hau_Latn	Hausa	Latin	Low
ibo_Latn	Igbo	Latin	Low
kir_Cyrl	Kyrgyz	Cyrillic	Low
npi_Deva	Nepali	Devanagari	Low
sin_Sinh	Sinhala	Sinhala	Low
som_Latn	Somali	Latin	Low
swh_Latn	Swahili	Latin	Low
tel_Telu	Telugu	Telugu	Low
yor_Latn	Yoruba	Latin	Low

Table 3: All 30 languages used in this paper with each language's corresponding ISO codes, scripts, and resource classifications defined by Singh et al. (2024)

A.2 Datasets

Easy Twenty Questions. We began with 200 English words from the dev and test sets of the *Things*[1] dataset (Zhang et al., 2024). We translated these words into all 30 target languages using Google Translate[2]. To ensure consistency and quality, we applied several filtering steps: we removed words where Latin characters persisted in non-Latin script languages, eliminated duplicates within each language, and filtered out remaining loan words to ensure semantic consistency across all languages. This filtering process yielded a final set of 140 words that maintained equivalence across all 30 languages. For each target word in each language, we randomly sampled 99 additional words from the same language to create a candidate pool of 100 words. The composition of these candidate pools and their ordering were kept consistent across all languages to ensure fair comparison. Table 4 provides example target words used in the Easy Twenty Questions task.

Other tasks and benchmarks. We utilized datasets available on Hugging Face for Belebele[3], HumanEvalExplain[4], Global-MMLU[5], and MultiQ[6]. Our experiments included the same 30 languages for Belebele and MultiQ that we used in our framework, while Global-MMLU experiments covered 29 languages (excluding Thai). For Global-MMLU, we specifically used only the Culturally-Agnostic (CA) subset to ensure fair cross-lingual comparability across all evaluated languages.

B Experimental Setup

B.1 Models

We conduct our evaluation by selecting recent LLMs, accessing with APIs. This information is summarized in Table 5.

B.2 Generations

The tasks used in our evaluation were configured with different generation parameters, such as temperature, token limits, and thresholds for fidelity scoring. Details for each task are provided in Table 6.

Generation settings. We modified several benchmark settings to ensure fair multilingual comparison. Key adjustments included explicitly prompting models to use the target language, rather than assuming responses would match the question language. For Code Reconstruction, we removed code description length limits since consistent length

[1] https://github.com/apple/ml-entity-deduction-arena

[2] https://translate.google.com

[3] https://hf.co/datasets/facebook/belebele

[4] https://hf.co/datasets/bigcode/humanevalpack

[5] https://hf.co/datasets/CohereLabs/Global-MMLU

[6] https://hf.co/datasets/caro-holt/MultiQ

	Translated Words		
ISO Code	Foam	Mango	Ice
amh_Ethi	አረፋ	ማንጎ	በረዶ
arb_Arab	رغوة	مانجو	ثلج
ben_Beng	ফোম	আম	বরফ
deu_Latn	Schaum	Mango	Eis
ell_Grek	Αφρός	Μάνγκο	Πάγος
eng_Latn	Foam	Mango	Ice
fra_Latn	Mousse	Mangue	Glace
hau_Latn	Kumfa	Mango	kankara
heb_Hebr	קצף	מנגו	קרח
hin_Deva	फोम	मैंगो	बर्फ़
ibo_Latn	ụfụfụ	Mango	ice
ind_Latn	Busa	Mangga	Es
ita_Latn	Schiuma	Mango	Ghiaccio
jpn_Jpan	泡	マンゴー	氷
kir_Cyrl	көбүк	Манго	Муз
kor_Hang	거품	망고	얼음
lit_Latn	Putos	Mangas	Ledas
npi_Deva	फोम	आँप	बरफ
por_Latn	Espuma	Manga	Gelo
ron_Latn	Spumă	Mango	Gheață
sin_Sinh	පෙන	අඹ	අයිස්
som_Latn	xumbo	Cambaha	baraf
spa_Latn	Espuma	Mango	Hielo
swh_Latn	Povu	Embe	barafu
tel_Telu	నురుగు	మామిడి	ఐస్
tha_Thai	โฟม	มะม่วง	น้ำแข็ง
ukr_Cyrl	Піна	Манго	Лід
yor_Latn	Foomu	Mango	Yinyin
zho_Hans	泡沫	芒果	冰
zsm_Latn	Buih	Mangga	Ais

Table 4: Example target words used in the Easy Twenty Questions task. Words were sourced from the *Things* dataset and translated into 30 languages via Google Translate.

constraints across different scripts isn't feasible. We use 5-shot prompting for Global-MMLU and zero-shot for Belebele.

Prompts. We provide prompts used for the three main tasks introduced in Section 3.1, as well as for established benchmarks which are Belebele (Bandarkar et al., 2024), MultiQ (Holtermann et al., 2024), and Global-MMLU (Singh et al., 2025) (for section §5.1). Each table outlines the role-specific prompts that we provided to two separate model instances. For Easy Twenty Questions and MCQ Conversation, the instances act as a *questioner* and an *answerer*; for Code Reconstruction, they act as a *describer* and a *rebuilder*. The prompt for Easy Twenty Questions is provided in Table 11, MCQ Conversation is in Table 12, and Code Reconstruction is in Table 13. The prompts for the preexisting three tasks are provided in Table 14.

Cost Analysis. The total cost to replicate our main results (Table 2) was approximately 608 USD, calculated using API pricing from OpenRouter and Azure OpenAI Service. The costs were distributed across the tasks as follows: Easy Twenty Questions (252 USD), MCQ Conversation (338 USD), and Code Reconstruction (18 USD). Notably, the evaluation of GPT-4o, our most expensive model, accounted for the majority of this expenditure at 449 USD.

C Detailed Experiment Results and Analysis

This section presents a comprehensive breakdown of our experimental results, including task-specific performance and its cross-lingual comparisons across multiple models. We also provide visualizations of task-wise correlations and additional evaluation results not included in the main paper.

C.1 Results on all languages on all models

Table 17, 18 present the evaluation results for all eight models across 30 languages and three tasks. For each model, we report task-wise accuracy scores across all languages, along with their corresponding Z-scores.

To account for varying task difficulties and enable a unified language ranking per model, we compute Z-scores that aggregate performance across the three tasks. Each task's scores are standardized independently, using the global mean and standard deviation computed over all models and languages for that task. This ensures that task-specific differences in difficulty are normalized appropriately. We then compute the average Z-score across the three tasks per language, allowing for relative performance comparisons across languages within each model.

A Z-score above 0 indicates that the model's accuracy on that language is above the global aver-

Model	Model Identifier	API Provider
GPT-4o	`gpt-4o-2024-08-06`	Azure OpenAI Service
GPT-4o-mini	`gpt-4o-mini-2024-07-18`	
Gemini-2.5-flash	`gemini-2.5-flash-preview-04-17`	OpenRouter
Gemini-2.0-flash	`gemini-2.0-flash-001`	
Qwen2.5-72B	`Qwen/Qwen2.5-72B-Instruct`	
Qwen2.5-7B	`Qwen/Qwen2.5-7B-Instruct`	
Llama-3.3-70B	`meta-llama/Llama-3.3-70B-Instruct`	
Llama-3.1-8B	`meta-llama/Llama-3.1-8B-Instruct`	

Table 5: Model identifiers and API providers used in experiments

Name	Temperature	Max Tokens	Fidelity Threshold
Easy Twenty Questions	0.7	Questioner: 1024 Answerer: 128	Language: 0.7 Answer: 0.9
MCQ conversation	0.7	Questioner: 2048 Answerer: 256	Language: 0.9 Answer: 0.9
Code Reconstruction	Describer: 0.7 Rebuilder: 0.2	2048	Language: 0.9
Global MMLU	0.0	32	N/A
Belebele	0.7	2048	N/A
MultiQ	0.0	Model: 256 Judge: 32	Language: 0.9

Table 6: Task-specific generation settings used in the evaluation

age, while a negative score suggests below-average performance. These aggregated Z-scores provide a normalized basis for ranking languages within each model and allow for interpretable comparisons.

C.2 Visualizations of task-wise correlations

We present a set of 6×6 scatter plots in Figure 10, visualizing pairwise correlations between the six tasks. Each plot compares the accuracy scores of two tasks across all 30 languages for 8 models, resulting in one point per language per model.

Each point in a scatter plot represents the performance of a particular language on two different tasks, with the x- and y-axes indicating the accuracy scores for each task. These visualizations help identify trends and clusters, revealing how performance on one task relates to another across languages.

These scatter plots serve as a visual counterpart to the Pearson correlation coefficients (r) reported in Figure 5, offering an intuitive understanding of inter-task relationships observed in our experiments.

C.3 Additional plot about language resource flexibility on MCQ Conversation

Following up on the analysis in Section 5.2, we conducted the same experiment with GPT-4o-mini under identical settings.

Figure 8 presents the MCQ Conversation accuracy across 30 languages when passages are provided in four different conditions: (1) the target language, (2) English, (3) a fixed set of five high-resource languages (averaged), and (4) a selection of up to five high-resource languages that are most similar to the target language. The overall trend is consistent with that of GPT-4o (Figure 6).

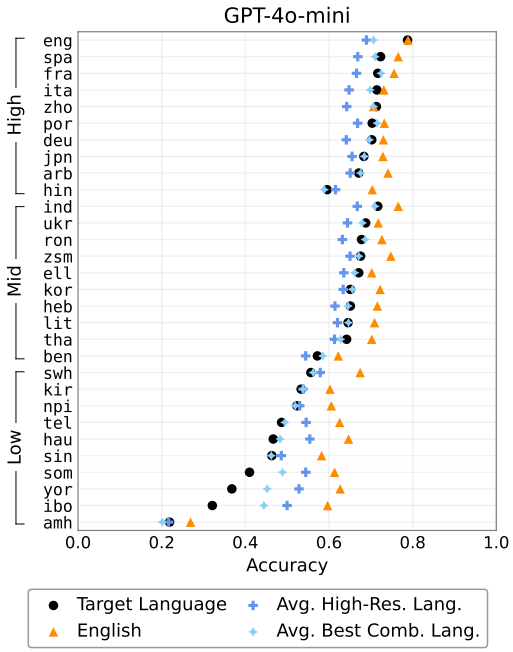

Figure 8: MCQ Conversation accuracy comparison across 30 languages for GPT-4o-mini, using passages in: (1) the target language, (2) English, (3) a fixed set of five high-resource languages (averaged), and (4) a selection of up to five high-resource languages most similar to the target language, with scores averaged.

C.4 Additional analysis about language resource flexibility on MCQ Conversation

To complement the substitution analysis in Section 5.2, Table 7 lists, for each of the 30 target languages, the subset of high-resource languages (selected from English, Chinese, Japanese, Hindi, and Arabic) that most closely approximates the original target-language passage in terms of MCQ Conversation accuracy.

The optimal subset for each target language was determined by selecting the combination (up to five languages) that minimizes the L2 distance from the original accuracy, as described in Section 5.2. When the target language itself was one of the five high-resource languages, it was excluded from its own substitution set. These exclusions are marked with ✗ in the corresponding table entries.

ISO Code	Language	Resources	ENG	ZHO	ARB	JPN	HIN
spa_Latn	Spanish	High	✔	✔	✔		
arb_Arab	Arabic	High	✔	✔	✗	✔	
deu_Latn	German	High	✔	✔			
fra_Latn	French	High	✔	✔			
ita_Latn	Italian	High	✔	✔			
por_Latn	Portuguese	High	✔	✔			
zho_Hans	Chinese	High	✔	✗			
eng_Latn	English	High	✗	✔			
jpn_Jpan	Japanese	High		✔		✗	
hin_Deva	Hindi	High				✔	✗
zsm_Latn	Malay	Mid	✔	✔	✔	✔	
lit_Latn	Lithuanian	Mid	✔	✔	✔		
kor_Hang	Korean	Mid	✔		✔	✔	
ben_Beng	Bengali	Mid	✔	✔			
ron_Latn	Romanian	Mid	✔	✔			
ukr_Cyrl	Ukrainian	Mid	✔		✔		
ell_Grek	Greek	Mid	✔			✔	
heb_Hebr	Hebrew	Mid	✔			✔	
ind_Latn	Indonesian	Mid	✔			✔	
tha_Thai	Thai	Mid	✔				✔
sin_Sinh	Sinhala	Low		✔	✔	✔	✔
npi_Deva	Nepali	Low	✔		✔		✔
kir_Cyrl	Kyrgyz	Low		✔	✔		
amh_Ethi	Amharic	Low			✔		
swh_Latn	Swahili	Low			✔		
hau_Latn	Hausa	Low					✔
ibo_Latn	Igbo	Low					✔
som_Latn	Somali	Low					✔
tel_Telu	Telugu	Low					✔
yor_Latn	Yoruba	Low					✔

Table 7: Optimal subsets of high-resource languages (selected from English, Chinese, Japanese, Hindi, and Arabic) for approximating the native-language passage performance in the MCQ Conversation task. For each target language, the listed subset scores the lowest L2 distance from the original accuracy. If the target language is one of the five high-resource options, it is excluded from its own substitution set, denoted with ✗

C.5 Human analysis case on MCQ Conversation Errors

As described in Section 5.3, we conducted a qualitative error analysis for both the Easy Twenty Questions and MCQ Conversation tasks. Specifically, we examined which conversational agent—the Questioner or the Answerer—was primarily responsible for task failure in each case. Tables 15 and 16 provide illustrative examples of typical errors for each role, along with our analysis of the underlying issues.

C.6 Correlation with Human Evaluation on MultiQ

To further validate MUG-Eval's effectiveness as a proxy for human evaluation, we conducted a human analysis for MultiQ dataset on 14 languages and 8 models.

C.6.1 Setup

To empirically validate MUG-Eval's automated scores against human judgments, we conduct a human evaluation study. Because the conversational logs from our framework are highly structured, we use the more open-ended MultiQ benchmark (Holtermann et al., 2024) to test whether MUG-Eval scores generalize as a reliable proxy for general-purpose text quality. This study evaluates outputs from the eight LLMs for a set of 15 questions sampled from the original 200 in the MultiQ benchmark. The evaluation spans 14 languages selected to cover high-resource (Arabic, Chinese, English, French, Hindi), mid-resource (Bengali, Indonesian, Korean, Malay, Thai), and low-resource (Amharic, Kyrgyz, Sinhala, Swahili) categories.

We recruit 12 annotators—primarily university students in South Korea—each a native speaker of their assigned language(s). Ten annotators cover a single language, while two bilingual annotators are responsible for two languages each (French/Arabic and Chinese/Malay). The same 15 questions are selected for all languages. To ensure the evaluation is both manageable and effective at differentiating model performance, we prioritize the most challenging questions based on a preliminary LLM-as-judge scoring using Gemini-2.5-flash. Before the main task, annotators are calibrated using a standardized set of English examples scored by the authors to ensure consistent judgment. Each participant evaluates the full set of generated responses for their language(s) in a two-hour session. Following a rubric adapted from Hada et al. (2024a), responses are scored on a 5-point Likert scale across three criteria: Linguistic Acceptability (fluency and naturalness), Output Content Quality (coherence and clarity), and Task Quality (how well the response addresses the question).

C.6.2 Result

For each question–answer set across 14 languages and 8 models, we first computed annotation scores for three metrics—Linguistic Acceptability, Output Content Quality, and Task Quality—and then averaged them to obtain a Total Average score per sample. These final annotation scores were subsequently averaged across samples for each language–model pair, yielding 112 aggregated scores (14 languages × 8 models). We then examined the correlation between these human evaluation scores (based on the three criteria) and task-specific scores from MUG-Eval as well as three existing multilingual benchmarks, all provided per language and model. Correlation statistics are reported in Figure 9.

The results show moderate to strong correlations between human judgments and MUG-Eval scores across tasks and metrics. This demonstrates that although MUG-Eval was originally designed for structured, information-gap tasks, its task completion–based scores generalize well to open-ended question answering in MultiQ. The strongest correlation was with Task Quality and the weakest with Linguistic Acceptability, reflecting MUG-Eval's focus on accurate information transfer rather than fluency. These findings suggest that MUG-Eval scores align well with human evaluation, though the three metrics are not fully independent.

Figure 9: Correlation analysis between human annotation on MultiQ data and six tasks consisting of MUG-Eval and existing multilingual benchmarks. Heatmaps show Pearson's r (left) and Spearman's ρ (right) correlation coefficients between human annotation and six tasks. All correlations exceed 0.4, demonstrating medium to strong consistency between human annotation with other six tasks, validating MUG-Eval's effectiveness as a multilingual evaluation framework.

C.7 Extending MUG-Eval to Summarization

To demonstrate the extensibility of the MUG-Eval framework beyond our initial three tasks, we implement a new summarization task based on the same "information-gap" paradigm underlying MUG-Eval's design.

C.7.1 Methodology

We employed the same 8 models and 30 languages from MUG-Eval. The summarization evaluation was conducted as follows:

Summarization-Length Limit Normalization:

Using the FLORES+ (Goyal et al., 2022) dataset, which primarily contains human-translated texts, we sampled 100 English sentences and retrieved their translations in 30 target languages. For each pair, we computed the ratio of character lengths by dividing the length of the translated sentence in a target language by the length of the corresponding English sentence, using Python's `len()` function. These ratios were then applied for length control in multilingual summarization.

Dataset: We sampled 100 articles from the QAGS (Wang et al., 2020) dataset, each originally in English. For each article, we generated 5 English question–answer pairs using GPT-4o-mini, with answers reflecting key factual entities.

Evaluation Process: For each model, language, and article, we followed this process:

1. A Summarizer LLM (target model) produced a summary of the article in the target language. Its language was verified using GlotLID, and the length was constrained to $(\texttt{Original English article length}) \times 0.5 \times (\texttt{language length ratio})$.
2. An Answerer LLM (target model) received only the target-language summary and the English questions, and generated answers in English.
3. The generated answers were compared to the gold answers using an LLM-as-Judge (GPT-4o-mini). Since both gold and generated answers were in English, this evaluation setup avoids translation-related bias.

C.7.2 Results

The average accuracy of the Answerer LLM across all models and languages is reported in Table 10, and its correlation score with six tasks consisting of MuG-Eval and existing multilingual benchmarks is reported in Table 8. This experiment shows that MuG-Eval can be readily extended to summarization while preserving its information-gap design, enabling scalable evaluation without references or human judgments. The results further exhibit moderate-to-strong correlations with the original MuG-Eval tasks, indicating that the framework captures a generalizable signal of multilingual generation quality.

D Generation Statistics

As stated in Section 5.4, we report detailed generation statistics in Table 9, averaged over models and language groups. Specifically, we measured the following:

- **Token Count and Sequence Length**: The number of tokens (# Token) and total character count (# Char) are computed from outputs generated in the target language by the questioner or the describer. The number of tokens were computed using the tokenizer associated with each model used in the experiments.
- **Language Fidelity**: Fidelity is measured as the percentage of questioner or describer outputs identified by GlotLID as matching the target language.
- **Instruction-Following of the Answerer**: Answerer Instruction-Following (A I-F) is defined for Easy Twenty Questions and MCQ Conversation as the proportion of answerer responses that strictly follow the output format ("yes," "no," and "maybe").
- **Interaction Length**: The number of question turns per interaction (# Turn) is reported for Easy Twenty Questions and MCQ Conversation, both of which are multi-turn tasks.

	Pearson	Spearman
Easy Twenty Questions	0.65	0.63
MCQ Conversation	0.65	0.61
Code Reconstruction	**0.79**	**0.74**
Global MMLU	0.68	0.68
Belebele	0.66	0.65
MultiQ	0.62	0.65

Table 8: Correlation score of summarization task with six tasks consisting of MuG-Eval and existing multilingual benchmarks. Overall correlation scores show high correlation, suggesting that the extension of MuG-Eval to other domains is plausible.

		Easy Twenty Questions					MCQ Conversation					Code Reconstruction		
		# Token	# Char.	Fidelity	A I-F	# Turn	# Token	# Char.	Fidelity	A I-F	# Turn	# Token	# Char.	Fidelity
Language	All	29.95	52.12	95.00	99.57	14.33	49.07	103.85	98.32	99.50	3.99	181.04	374.30	97.72
	ENG	11.19	45.28	96.52	100.00	14.32	23.22	111.43	99.88	99.95	4.05	93.50	412.86	99.63
	High	16.19	44.47	95.78	99.30	14.25	30.24	94.63	97.76	99.37	3.98	113.36	341.29	97.91
	Mid	18.88	41.97	95.59	99.53	14.32	38.16	92.54	98.93	99.72	3.95	147.26	344.32	98.64
	Low	54.77	69.87	93.61	99.88	14.42	78.70	124.31	98.28	99.40	4.04	282.50	437.31	96.61
Model	GPT-4o	14.80	38.52	97.16	100.00	13.96	27.10	71.54	99.80	100.00	4.02	123.68	345.46	99.91
	Gemini-2.0-flash	9.81	22.60	94.49	99.99	15.59	44.15	110.25	99.06	100.00	4.21	124.75	332.00	99.85
	Gemini-2.5-flash	9.70	24.23	95.28	99.90	14.02	55.10	178.68	91.88	99.88	3.95	117.67	296.46	96.48
	Qwen2.5-72B	57.47	78.58	96.48	100.00	14.24	61.42	98.82	99.85	100.00	3.94	288.51	494.24	99.46
	GPT-4o-mini	14.28	34.67	97.64	100.00	16.00	47.45	81.39	99.85	100.00	4.08	124.70	351.47	99.98
	Llama-3.3-70B	38.33	82.68	91.52	99.93	11.07	46.92	82.72	99.86	98.85	4.00	139.93	256.66	99.83
	Qwen2.5-7B	61.22	81.07	93.78	99.83	16.50	77.60	128.97	97.16	99.92	3.30	256.50	443.95	92.84
	Llama-3.1-8B	33.83	54.97	93.62	96.89	13.25	87.03	138.48	99.12	97.34	4.40	272.59	474.21	93.39

Table 9: Average token count (# Token), character-level sequence length (# Character), GlotLID-based language fidelity (Fidelity), instruction-following rate of the answerer (A I-F), and average number of question turns (# Turn) are computed per task, model, and language group.

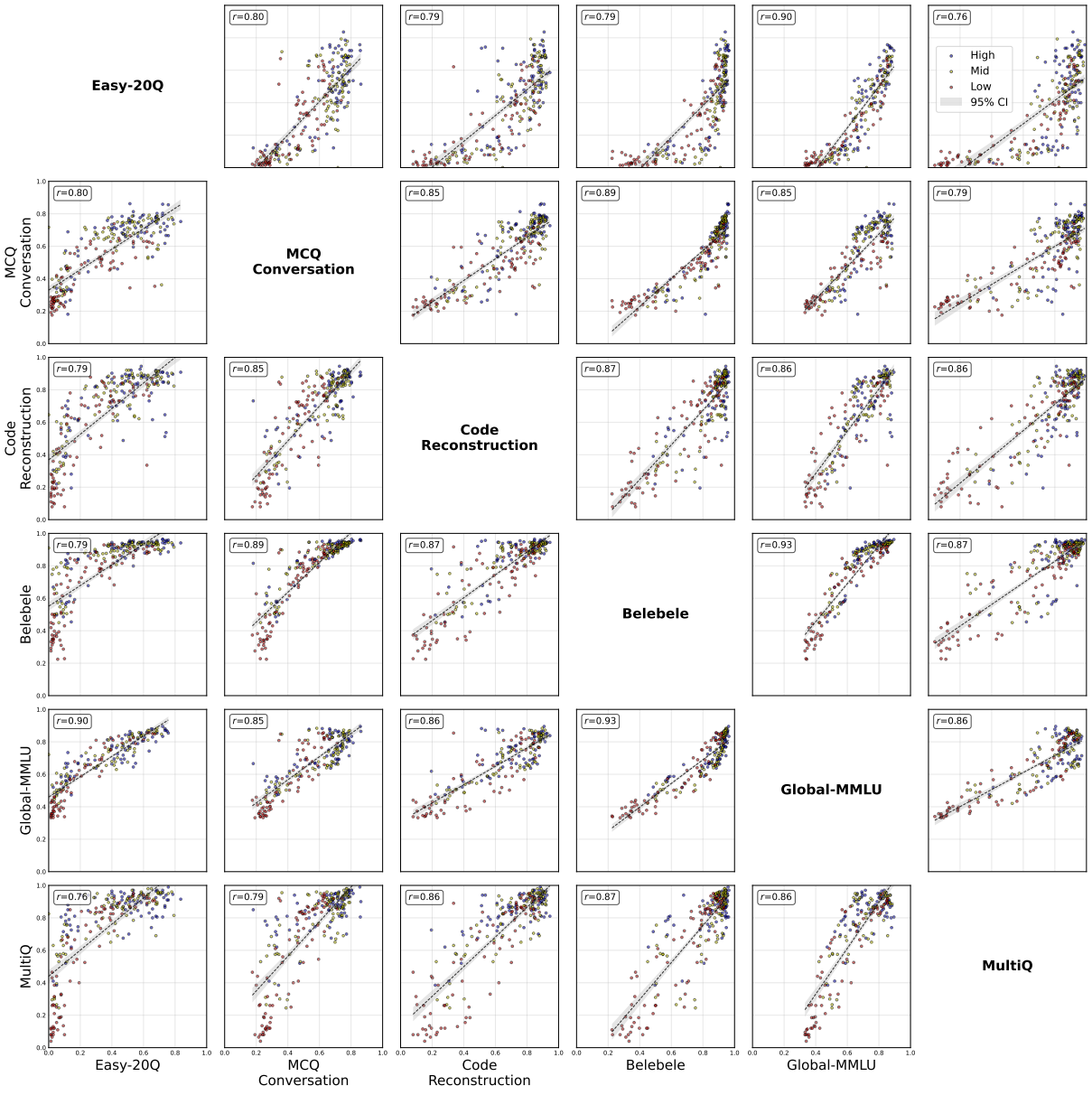

Figure 10: Correlation matrix showing relationships between MUG-Eval tasks and existing multilingual benchmarks. Each cell displays Pearson's correlation coefficient (r) with 95% confidence intervals, with points colored by language resource level.

	gpt-4o	gpt-4o-mini	gemini-2.5-flash	gemini-2.0-flash	qwen-2.5-72b	qwen-2.5-7b	llama-3.3-70b	llama-3.1-8b
Spanish	0.76	0.77	0.78	**0.79**	0.81	0.62	**0.52**	0.4
Arabic	0.57	0.56	0.54	0.53	0.57	0.3	0.25	0.07
German	0.76	**0.78**	0.8	0.75	0.81	0.55	0.49	0.39
French	0.76	0.76	0.79	0.76	0.8	0.6	0.46	**0.46**
Italian	0.76	0.75	0.8	0.78	0.8	0.6	0.45	0.34
Portuguese	**0.79**	0.77	0.78	0.74	0.82	0.57	0.5	0.39
Chinese	0.56	0.56	0.6	0.63	0.75	0.56	0.36	0.18
English	**0.79**	0.76	**0.83**	0.74	0.8	**0.64**	**0.52**	0.42
Japanese	0.58	0.52	0.6	0.65	0.71	0.47	0.37	0.11
Hindi	0.6	0.64	0.63	0.58	0.61	0.45	0.4	0.27
Malay	0.73	**0.78**	0.79	0.75	0.69	0.29	0.44	0.25
Lithuanian	0.71	0.71	0.71	0.67	0.71	0.47	0.42	0.33
Korean	0.6	0.52	0.6	0.68	0.62	0.37	0.38	0.13
Bengali	0.69	0.68	0.68	0.72	0.72	0.42	0.45	0.3
Romanian	0.76	**0.78**	0.78	0.76	0.8	0.54	0.5	0.39
Ukrainian	0.69	0.73	0.75	0.71	0.73	0.45	0.44	0.26
Greek	0.73	0.69	0.78	0.73	0.72	0.44	0.47	0.06
Hebrew	0.67	0.63	0.68	0.67	0.51	0.26	0.43	0.22
Indonesian	0.77	0.77	0.78	0.73	**0.85**	0.6	0.5	0.38
Thai	0.73	0.65	0.72	0.74	0.73	0.49	0.42	0.08
Sinhala	0.52	0.47	0.69	0.64	0.5	0.28	0.36	0.11
Nepali	0.68	0.65	0.74	0.74	0.64	0.44	0.42	0.1
Kyrgyz	0.65	0.65	0.7	0.72	0.61	0.4	0.34	0.11
Amharic	0.35	0.35	0.53	0.58	0.42	0.21	0.31	0.18
Swahili	0.75	0.75	0.79	0.75	0.65	0.41	0.42	0.33
Hausa	0.74	0.77	0.77	0.72	0.66	0.4	0.45	0.3
Igbo	0.71	**0.78**	0.8	0.73	0.64	0.42	0.45	0.26
Somali	0.73	0.7	0.74	0.74	0.63	0.36	0.46	0.28
Telugu	0.65	0.61	0.7	0.69	0.57	0.33	0.41	0.33
Yoruba	0.67	0.71	0.78	0.71	0.63	0.44	0.5	0.26

Table 10: Results of the summarization task: average accuracy of the Answerer LLM across 8 models and 30 languages, demonstrating the extensibility of MUG-Eval framework

Role	Type	Prompt
Questioner	Initial Instruction	You will be solving an entity deduction game by asking questions about a hidden item in {lang_full}. Your goal is to identify exactly one correct entity from a list of 100 items through strategic questioning, using as few questions as possible. You can ask yes/no/maybe questions in {lang_full}, one at a time. Each question must be concise and answerable only with "Yes," "No," or "Maybe." Do not ask for hints or the answer directly. Among the list, exactly one is correct. You have a maximum of 20 questions before making your final choice, but fewer questions are better. When you have determined the correct entity, provide your final answer using double brackets: [[entity]]. The entity must be exactly as it appears in the Entity List (do not translate to another language). <|Entity List|> {entity_list} <|Language|> {lang_full} Begin with your first question.
Answerer	Instruction	"Answer the following question about the hidden entity: $\{entity\}$. Respond only with ""Yes,"" ""No,"" or ""Maybe"" (including the period) without providing explanations or additional words. Do not use ""Maybe"" if the answer can be definitively ""Yes"" or ""No."" Your responses must be in English. Never mention the entity in your response. If the query contains two or more questions asked simultaneously, consider it invalid and respond with ""Maybe."" <|Entity|> $\{entity\}$ <|Language|> $\{lang\_full\}$ <|Question|> $\{question\}$ <|Options|> - Yes. - No. - Maybe.
Questioner	Final Instruction After 20 Turns	You have now used all available questions. Based on the responses you've received, provide your final guess of the entity. Even if you cannot determine the entity with confidence, provide your best guess based on the information gathered. Indicate your answer in double brackets: [[entity]]. The entity must be exactly as it appears in the Entity List (do not translate to another language).

Table 11: Prompt design for the Easy Twenty Questions task. The questioner and answerer are separately prompted with role-specific instructions to simulate a Twenty Questions game. The prompts include task rules, language constraints, response formatting requirements, and structured input fields (*e.g.*, `{entity_list}`, `{lang_full}`, `{question}`)

Role	Type	Prompt
Questioner	Initial Instruction	You will be solving a multiple-choice question by asking questions about a hidden passage. I am serving as an intermediary between you and a person who has this passage. You can ask me questions about the passage content, and I will relay these to the person. They will respond with only "yes," "no," or "maybe," which I will then share with you. Your questions must be in {lang_full} and you can only ask one question at a time. Do not ask for hints or request the passage directly. Among the four answer choices provided, exactly one is correct. You must ask exactly 4 questions (one corresponding to each answer choice) before making your final decision. After receiving all four responses, provide your final answer in {lang_full}, indicating the correct number choice with double brackets: [[X]] <|Query|> {query} <|Choices|> (1) {c1} (2) {c2} (3) {c3} (4) {c4} <|Language|> {lang_full}
Answerer	Instruction	Answer the following question based solely on the information contained in the passage. Do not use any external knowledge or make assumptions beyond what is explicitly stated in the text. Your response must be in English and limited to exactly one of these three answers: "Yes.", "No.", or "Maybe." (including the period). Do not provide any explanation or additional words. If the query contains two or more questions asked simultaneously, consider it invalid and respond with "Maybe." <|Passage|> passage <|Language|> lang_full <|Question|> question <|Options|> - Yes. - No. - Maybe.
Questioner	Final Instruction After 10 Turns	You have now used all available questions. Based on the responses you've received, please provide your final answer in lang_full. Indicate your choice by including the number of the correct answer in double brackets: [[X]]

Table 12: Prompt design for the MCQ Conversation task. The questioner and answerer are assigned separate prompts to simulate a collaborative multiple-choice reasoning task. The questioner asks yes/no questions based on a hidden passage, while the answerer responds with constrained answers. Prompts include language and formatting instructions, as well as structured fields such as `{query}`, `{passage}`, and `{lang_full}`.

Role	Type	Prompt
Describer	Instruction for the English Language	Provide a concise natural language description of the code in {lang_full}. <|Code|> {code} <|Language|> {lang_full}
Describer	Instruction for Non-English Languages	Provide a concise natural language description of the code in lang_full. Variable names or technical terms in English are allowed, but there shouldn't be any phrases or full sentences written in English or any language except lang_full. <|Code|> code <|Language|> lang_full
Rebuilder	Instruction	"Write functional code in {$code\_lang\_full$} according to the description written in {$lang\_full$}. Start your code with the given declaration. Write only the code without any additional descriptions, comments, or delimiters. <|Language|> {$lang\_full$} <|Description|> {$description$} <|Declaration|> {$declaration$}

Table 13: Prompt design for the Code Reconstruction task. One model instance (the describer) generates a natural language description of a given code snippet in the target language. Another instance (the rebuilder) reconstructs the original function from this description and a given declaration. The prompts specify language requirements and restrict the output format to code-only.

Benchmark	Type	Prompt
MultiQ	Evaluatee	Please provide an answer to the following question in {lang_full}. <|Question|> {question} <|Language|> {lang_full}
MultiQ	Classifier (LLM-as-a-Judge)	Determine if the answer to the following question is correct. The question is in English and the answer is in {lang_full}. Respond only with 'yes' or 'no' - do not include explanations or additional words. <|Question|> {question_en} <|Language|> {lang_full} <|Answer|> {model_pred}
Global-MMLU	-	{question} A. {option_a} B. {option_b} C. {option_c} D. {option_d} Answer:
Belebele	-	Given the following passage, query, and answer choices, output the number corresponding to the correct answer in double brackets: [[X]] <|Language|> {lang_full} <|Passage|> {passage} <|Query|> query <|Choices|> (1) {c1} (2) {c2} (3) {c3} (4) {c4}

Table 14: Prompt design for the pre-existing benchmark tasks used in our evaluation. For MultiQ, we include both evaluatee prompts and classification prompts for LLM-as-a-Judge. Global-MMLU and Belebele use simpler one-shot prompts formatted according to their original task definitions. Prompts include structured input fields such as `{question}`, `{lang_full}`, and `{choices}`

Language	Korean
Passage	Victoria Falls is a small city in western Zimbabwe, across the border from Livingston, Zambia and Botswana. This town is located right next to the waterfalls, and they are the town's main attraction, and this popular tourist attraction also offers many opportunities for adventurers and tourists alike to stay longer. During the rainy season (November to March), waterfalls increase in volume, and the waterfalls become more dramatic. Crossing a bridge or walking down a winding path near the waterfall will cause your clothes to get wet. On the other hand, the amount of water is so large that the actual size of the waterfall is obscured by the sheer volume! 빅토리아 폴스는 리빙스턴, 잠비아와 보츠와나 근처의 국경 건너편의 짐바브웨 서쪽에 있는 작은 도시입니다. 이 마을은 폭포 바로 옆에 위치해 있고, 그 폭포들이 마을의 주요 명소이며, 또한 이 유명한 관광지는 모험가들과 관광객들 모두에게 더 오래 머물 수 있는 많은 기회를 제공한다. 우기(11월부터 3월까지)에는 수량이 증가하며, 폭포가 더 극적으로 변합니다. 다리를 건너거나 폭포 근처의 구불구불한 길을 따라 걸어가면 옷이 젖게 됩니다. 반면에, 물의 양이 너무 많아서 실제 보이는 폭포의 크기는 - 엄청난 수량으로 인해 - 가려진다!
Query	If I want an open view , when should I visit the waterfall? 탁 트인 전망을 원한다면 언제 폭포를 방문해야 합니까?
Choices	1. From November to March 11월부터 3월까지 2. During the tourist season 관광철에 3. When the water volume is high 수량이 많을 때 4. When it's not rainy season 장마철이 아닐 때
Conversation	Questioner: Is the period from November to March suitable for visiting the waterfall? 폭포를 방문하기에 11월부터 3월까지가 적합한가요? Answerer: Yes. Questioner: Is the tourist season suitable for visiting the waterfall? 폭포를 방문하기에 관광철이 적합한가요? Answerer: Maybe. Questioner: Is it suitable to visit the waterfall when the water volume is high? 폭포를 방문하기에 수량이 많을 때가 적합한가요? Answerer: Maybe. Questioner: Is it suitable to visit the waterfall when it is not the rainy season? 폭포를 방문하기에 장마철이 아닐 때가 적합한가요? Answerer: Maybe.
Error Analysis	Each questions from the Questioner **does not ask** whether the visiting timing is suitable if the visitor wants an open view . As a result, each question becomes *an ambiguous question* that is difficult to answer based on the passage.

Table 15: Human analysis of a case where the Questioner made an erroneous output in MCQ Conversation, ran by `gpt-4o-2024-08-06`. The original dataset and conversation are in Korean, with the Korean text shown in gray.

Language	English
Passage	During the 18th century Cambodia found itself squeezed between two powerful neighbors, Thailand and Vietnam. The Thais invaded Cambodia several times in the 18th century and in 1772 they destroyed Phnom Phen. In the last years of the 18th century the Vietnamese also invaded Cambodia.
Query	Which of the following statements accurately describes the invasions of Cambodia?
Choices	1. The Vietnamese invaded Cambodia several times 2. The Thais and Vietnamese invaded Cambodia in 1772 3. The Vietnamese destroyed Phnom Phen 4. The Thais invaded Cambodia several times
Conversation	Questioner: Is it true that the Vietnamese invaded Cambodia several times? Answerer: No. Questioner: Did the Thais and Vietnamese invade Cambodia in 1772? Answerer: Yes. Questioner: Is it true that the Vietnamese destroyed Phnom Penh? Answerer: No. Questioner: Did the Thais invade Cambodia several times? Answerer: Yes.
Error Analysis	The answerer gave an incorrect response to the second question. While there is a mention that the Thais invaded in 1772, there is *no information indicating that the Vietnamese invaded Cambodia in 1772*. However, the answerer responded Yes. to the second question.

Table 16: Human analysis of a case where the Answerer made an erroneous output in MCQ Conversation, ran by `gpt-4o-2024-08-06`.

gpt-4o-2024-08-06

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	1.37	1.62 (75.7)	1.56 (85.6)	0.94 (88.4)
high	zho	1.33	1.94 (83.6)	1.01 (75.2)	1.04 (90.9)
mid	ron	1.29	1.7 (77.9)	1.14 (77.7)	1.02 (90.2)
high	ita	1.26	1.73 (78.6)	1.07 (76.3)	0.99 (89.6)
high	por	1.24	1.47 (72.1)	1.21 (79.1)	1.04 (90.9)
high	spa	1.18	1.29 (67.9)	1.16 (78.1)	1.09 (92.1)
mid	ell	1.17	1.35 (69.3)	1.08 (76.6)	1.07 (91.5)
high	fra	1.16	1.32 (68.6)	1.18 (78.4)	0.99 (89.6)
mid	ukr	1.16	1.62 (75.7)	1.09 (76.7)	0.79 (84.8)
mid	heb	1.13	1.76 (79.3)	0.93 (73.8)	0.69 (82.3)
high	arb	1.12	1.29 (67.9)	1.11 (77.1)	0.97 (89)
high	deu	1.12	1.32 (68.6)	1.27 (80.2)	0.76 (84.1)
high	jpn	1.08	1.5 (72.9)	0.95 (74.1)	0.79 (84.8)
mid	zsm	1.06	1.08 (62.9)	1.12 (77.3)	0.97 (89)
mid	ind	1.05	1.17 (65)	1.1 (77)	0.86 (86.6)
mid	kor	1.05	1.35 (69.3)	1.07 (76.4)	0.71 (82.9)
high	hin	1.04	1.41 (70.7)	0.67 (68.9)	1.04 (90.9)
mid	tha	1.03	1.41 (70.7)	0.83 (71.8)	0.86 (86.6)
mid	ben	0.84	1.35 (69.3)	0.24 (60.7)	0.94 (88.4)
mid	lit	0.81	0.67 (52.9)	1.02 (75.4)	0.74 (83.5)
low	kir	0.77	1.05 (62.1)	0.68 (69)	0.59 (79.9)
low	npi	0.72	1.05 (62.1)	0.36 (63)	0.74 (83.5)
low	swh	0.68	0.2 (41.4)	0.94 (73.9)	0.92 (87.8)
low	tel	0.56	0.7 (53.6)	0.3 (61.8)	0.69 (82.3)
low	sin	0.35	0.34 (45)	0.4 (63.7)	0.31 (73.2)
low	som	0.16	0.17 (40.7)	-0.01 (55.9)	0.33 (73.8)
low	hau	0.1	-0.28 (30)	0.12 (58.4)	0.46 (76.8)
low	yor	-0.01	0.2 (41.4)	-0.49 (46.8)	0.28 (72.6)
low	ibo	-0.07	0.08 (38.6)	-0.57 (45.3)	0.28 (72.6)
low	amh	-0.46	-0.16 (32.9)	-0.32 (50)	-0.89 (44.5)

gpt-4o-mini-2024-07-18

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	0.94	0.7 (53.6)	1.2 (78.8)	0.92 (87.8)
high	zho	0.77	0.76 (55)	0.8 (71.3)	0.74 (83.5)
mid	ukr	0.68	0.55 (50)	0.67 (68.8)	0.81 (85.4)
high	spa	0.66	0.37 (45.7)	0.86 (72.3)	0.74 (83.5)
high	fra	0.59	0.2 (41.4)	0.82 (71.7)	0.76 (84.1)
high	por	0.58	0.14 (40)	0.75 (70.3)	0.86 (86.6)
high	deu	0.57	0.46 (47.9)	0.75 (70.2)	0.51 (78)
high	jpn	0.54	0.4 (46.4)	0.65 (68.3)	0.56 (79.3)
mid	ron	0.51	0.14 (40)	0.62 (67.8)	0.79 (84.8)
high	hin	0.46	0.7 (53.6)	0.18 (59.6)	0.51 (78)
mid	ell	0.45	0.2 (41.4)	0.58 (67.1)	0.59 (79.9)
high	ita	0.39	-0.31 (29.3)	0.81 (71.4)	0.66 (81.7)
mid	kor	0.37	-0.01 (36.4)	0.48 (65.1)	0.64 (81.1)
mid	zsm	0.34	-0.25 (30.7)	0.6 (67.6)	0.66 (81.7)
high	arb	0.33	-0.28 (30)	0.58 (67.1)	0.69 (82.3)
mid	tha	0.29	-0.07 (35)	0.43 (64.2)	0.51 (78)
mid	ind	0.29	-0.57 (22.9)	0.82 (71.7)	0.61 (80.5)
mid	ben	0.28	0.25 (42.9)	0.06 (57.2)	0.53 (78.7)
mid	heb	0.22	-0.22 (31.4)	0.48 (65.1)	0.41 (75.6)
mid	lit	0.17	-0.34 (28.6)	0.45 (64.6)	0.41 (75.6)
low	npi	0	-0.13 (33.6)	-0.2 (52.3)	0.33 (73.8)
low	kir	-0.1	-0.19 (32.1)	-0.15 (53.3)	0.03 (66.5)
low	swh	-0.16	-0.96 (13.6)	-0.02 (55.7)	0.51 (78)
low	tel	-0.28	-0.66 (20.7)	-0.39 (48.7)	0.21 (70.7)
low	hau	-0.53	-1.02 (12.1)	-0.5 (46.7)	-0.07 (64)
low	som	-0.61	-1.17 (8.6)	-0.8 (41)	0.13 (68.9)
low	sin	-0.61	-1.05 (11.4)	-0.52 (46.3)	-0.28 (59.1)
low	ibo	-0.77	-0.93 (14.3)	-1.27 (32.1)	-0.12 (62.8)
low	yor	-0.92	-1.2 (7.9)	-1.02 (36.8)	-0.53 (53)
low	amh	-1.61	-1.43 (2.1)	-1.81 (21.9)	-1.6 (27.4)

gemini-2.5-flash-preview

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	1.36	1.47 (72.1)	1.57 (85.9)	1.04 (90.9)
high	zho	1.15	1.59 (75)	0.85 (72.2)	1.02 (90.2)
mid	ukr	1.12	1.29 (67.9)	1.02 (75.3)	1.07 (91.5)
mid	ind	1.11	1.32 (68.6)	1.19 (78.6)	0.81 (85.4)
mid	heb	1.07	1.29 (67.9)	1 (75.1)	0.92 (87.8)
high	arb	1.02	1.32 (68.6)	0.96 (74.2)	0.79 (84.8)
mid	zsm	1.02	1.11 (63.6)	1.22 (79.2)	0.71 (82.9)
high	hin	0.98	1.62 (75.7)	0.55 (66.6)	0.76 (84.1)
high	jpn	0.92	1.05 (62.1)	0.82 (71.7)	0.89 (87.2)
mid	kor	0.87	1.23 (66.4)	0.64 (68.2)	0.74 (83.5)
mid	tha	0.84	1.35 (69.3)	0.11 (58.1)	1.07 (91.5)
mid	ron	0.81	1.32 (68.6)	0.12 (58.3)	0.99 (89.6)
low	sin	0.8	1.2 (65.7)	0.31 (61.9)	0.89 (87.2)
high	spa	0.74	1.17 (65)	0.55 (66.4)	0.51 (78)
high	ita	0.74	1.53 (73.6)	0.74 (70.1)	-0.05 (64.6)
low	tel	0.63	1.11 (63.6)	0.34 (62.6)	0.43 (76.2)
mid	lit	0.56	0.85 (57.1)	1 (75)	-0.15 (62.2)
high	deu	0.54	1.56 (74.3)	0.56 (66.7)	-0.51 (53.7)
mid	ell	0.53	1.08 (62.9)	-0.15 (53.3)	0.66 (81.7)
mid	ben	0.43	1.44 (71.4)	-1.05 (36.2)	0.92 (87.8)
high	fra	0.41	1.53 (73.6)	0.3 (61.8)	-0.61 (51.2)
low	amh	0.36	0.31 (44.3)	0.47 (65)	0.31 (73.2)
low	npi	0.3	1.26 (67.1)	-1.1 (35.3)	0.74 (83.5)
low	swh	0.3	0.11 (39.3)	0.36 (62.9)	0.43 (76.2)
high	por	0.3	1.2 (65.7)	-0.14 (53.4)	-0.18 (61.6)
low	ibo	0.27	0.4 (46.4)	-0.36 (49.3)	0.76 (84.1)
low	kir	-0.04	1.05 (62.1)	0.18 (59.4)	-1.34 (33.5)
low	som	-0.04	0.17 (40.7)	-0.28 (50.9)	-0.02 (65.2)
low	yor	-0.06	0.11 (39.3)	-0.53 (46.1)	0.26 (72)
low	hau	-0.19	-0.28 (30)	-0.47 (47.2)	0.18 (70.1)

gemini-2.0-flash-001

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	1.06	0.61 (51.4)	1.59 (86.2)	0.97 (89)
high	fra	1.04	0.88 (57.9)	1.13 (77.6)	1.12 (92.7)
high	jpn	1	1.2 (65.7)	1.02 (75.4)	0.79 (84.8)
mid	ukr	0.99	0.96 (60)	1.05 (76)	0.97 (89)
high	zho	0.99	0.91 (58.6)	0.87 (72.7)	1.19 (94.5)
mid	ron	0.96	0.67 (52.9)	1.19 (78.6)	1.02 (90.2)
high	por	0.93	0.64 (52.1)	1.04 (75.9)	1.09 (92.1)
mid	lit	0.91	0.58 (50.7)	1.06 (76.1)	1.09 (92.1)
mid	zsm	0.88	0.91 (58.6)	0.85 (72.1)	0.89 (87.2)
mid	ind	0.88	0.67 (52.9)	0.99 (74.9)	0.97 (89)
high	hin	0.87	0.99 (60.7)	0.64 (68.2)	0.97 (89)
high	deu	0.86	0.58 (50.7)	1.03 (75.6)	0.97 (89)
high	spa	0.84	0.91 (58.6)	0.66 (68.7)	0.94 (88.4)
mid	kor	0.8	0.82 (56.4)	0.63 (68)	0.97 (89)
mid	ell	0.79	0.91 (58.6)	0.46 (64.8)	1.02 (90.2)
mid	heb	0.74	0.7 (53.6)	0.63 (68.1)	0.89 (87.2)
mid	tha	0.72	0.52 (49.3)	0.54 (66.3)	1.09 (92.1)
high	ita	0.69	0.49 (48.6)	0.58 (67.1)	0.99 (89.6)
high	arb	0.68	0.82 (56.4)	0.52 (66)	0.71 (82.9)
mid	ben	0.61	1.08 (62.9)	-0.19 (52.6)	0.94 (88.4)
low	sin	0.6	0.96 (60)	0.36 (63)	0.46 (76.8)
low	kir	0.57	0.61 (51.4)	0.55 (66.6)	0.56 (79.3)
low	tel	0.57	0.7 (53.6)	0.36 (63)	0.64 (81.1)
low	swh	0.5	-0.07 (35)	0.8 (71.2)	0.76 (84.1)
low	amh	0.34	0.76 (55)	-0.16 (53)	0.43 (76.2)
low	hau	0.26	-0.43 (26.4)	0.28 (61.3)	0.92 (87.8)
low	som	0.23	-0.04 (35.7)	0.04 (56.9)	0.69 (82.3)
low	yor	0.15	-0.13 (33.6)	-0.23 (51.7)	0.81 (85.4)
low	ibo	0.11	0.11 (39.3)	-0.33 (49.8)	0.56 (79.3)
low	npi	0.08	0.61 (51.4)	-1.15 (34.3)	0.79 (84.8)

Table 17: Results for each task on MUG-Eval across 30 languages, evaluated using gpt-4o-2024-08-06, gpt-4o-mini-2024-07-18, gemini-2.5-flash-preview, and gemini-2.0-flash-001. Accuracy was normalized using Z-scores and averaged across tasks. Languages were then ranked by their averaged Z-score.

llama-3.3-70b-instruct

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	0.81	0.7 (53.6)	1.33 (81.3)	0.41 (75.6)
high	zho	0.79	1.11 (63.6)	0.94 (73.9)	0.31 (73.2)
high	fra	0.56	0.61 (51.4)	0.96 (74.3)	0.1 (68.3)
mid	ind	0.55	0.73 (54.3)	0.99 (74.9)	-0.07 (64)
high	spa	0.54	0.43 (47.1)	0.9 (73.1)	0.28 (72.6)
mid	ron	0.53	0.61 (51.4)	0.76 (70.6)	0.21 (70.7)
mid	ukr	0.51	0.55 (50)	0.8 (71.3)	0.18 (70.1)
mid	zsm	0.5	0.73 (54.3)	0.87 (72.6)	-0.1 (63.4)
high	por	0.45	0.14 (40)	0.92 (73.4)	0.31 (73.2)
high	ita	0.4	0.25 (42.9)	0.77 (70.8)	0.18 (70.1)
high	deu	0.36	0.14 (40)	0.8 (71.2)	0.15 (69.5)
mid	ell	0.24	0.05 (37.9)	0.51 (65.8)	0.15 (69.5)
mid	heb	0.2	0.28 (43.6)	0.37 (63.1)	-0.05 (64.6)
mid	tha	0.14	0.17 (40.7)	0.46 (64.9)	-0.2 (61)
high	arb	0.11	0.43 (47.1)	0.6 (67.4)	-0.71 (48.8)
mid	lit	0.1	-0.31 (29.3)	0.66 (68.7)	-0.05 (64.6)
mid	ben	0.1	0.23 (42.1)	0.18 (59.4)	-0.1 (63.4)
high	jpn	-0.08	-1.02 (12.1)	0.67 (68.9)	0.1 (68.3)
high	hin	-0.15	0.28 (43.6)	-0.53 (46)	-0.2 (61)
low	tel	-0.23	-0.54 (23.6)	0.15 (58.9)	-0.3 (58.5)
low	swh	-0.24	-0.43 (26.4)	0.37 (63.1)	-0.66 (50)
mid	kor	-0.25	-1.52 (0)	0.82 (71.7)	-0.05 (64.6)
low	npi	-0.38	-0.13 (33.6)	-0.43 (48)	-0.58 (51.8)
low	kir	-0.54	-0.84 (16.4)	-0.04 (55.4)	-0.73 (48.2)
low	sin	-0.58	-0.78 (17.9)	0.18 (59.4)	-1.14 (38.4)
low	hau	-0.86	-0.96 (13.6)	-0.5 (46.7)	-1.11 (39)
low	ibo	-1.08	-0.87 (15.7)	-1.04 (36.4)	-1.34 (33.5)
low	som	-1.26	-1.2 (7.9)	-1.23 (32.9)	-1.37 (32.9)
low	yor	-1.35	-1.11 (10)	-1.36 (30.3)	-1.57 (28)
low	amh	-1.68	-1.37 (3.6)	-1.91 (20)	-1.75 (23.8)

llama-3.1-8b-instruct

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	-0.56	-0.49 (25)	-0.63 (44.2)	-0.58 (51.8)
high	spa	-0.62	-0.84 (16.4)	-0.87 (39.6)	-0.15 (62.2)
high	ita	-0.85	-0.99 (12.9)	-1.19 (33.6)	-0.38 (56.7)
high	por	-0.96	-1.2 (7.9)	-1.24 (32.6)	-0.43 (55.5)
mid	ind	-0.97	-1.05 (11.4)	-1.13 (34.8)	-0.73 (48.2)
high	fra	-0.99	-0.96 (13.6)	-1.55 (26.8)	-0.45 (54.9)
high	deu	-1.1	-1.08 (10.7)	-1.61 (25.7)	-0.61 (51.2)
high	zho	-1.12	-1.05 (11.4)	-1.23 (32.9)	-1.09 (39.6)
mid	ukr	-1.13	-1.08 (10.7)	-1.27 (32)	-1.04 (40.9)
mid	ron	-1.14	-1.17 (8.6)	-1.4 (29.6)	-0.86 (45.1)
high	jpn	-1.17	-1.14 (9.3)	-1.26 (32.2)	-1.11 (39)
mid	zsm	-1.21	-1.14 (9.3)	-1.21 (33.2)	-1.29 (34.8)
mid	kor	-1.21	-1.22 (7.1)	-1.15 (34.3)	-1.27 (35.4)
high	hin	-1.24	-0.93 (14.3)	-1.22 (33)	-1.57 (28)
mid	tha	-1.26	-1.14 (9.3)	-1.3 (31.4)	-1.34 (33.5)
mid	ell	-1.3	-1.28 (5.7)	-1.56 (26.7)	-1.06 (40.2)
mid	lit	-1.35	-1.31 (5)	-1.29 (31.7)	-1.44 (31.1)
high	arb	-1.43	-1.31 (5)	-1.16 (34.1)	-1.82 (22)
mid	heb	-1.49	-1.37 (3.6)	-1.6 (25.9)	-1.49 (29.9)
mid	ben	-1.6	-1.25 (6.4)	-1.76 (22.8)	-1.8 (22.6)
low	ibo	-1.61	-1.14 (9.3)	-1.71 (23.7)	-1.98 (18.3)
low	swh	-1.63	-1.43 (2.1)	-1.54 (26.9)	-1.9 (20.1)
low	som	-1.64	-1.28 (5.7)	-1.57 (26.4)	-2.08 (15.9)
low	hau	-1.65	-1.2 (7.9)	-1.66 (24.7)	-2.1 (15.2)
low	tel	-1.7	-1.31 (5)	-1.87 (20.8)	-1.93 (19.5)
low	kir	-1.75	-1.37 (3.6)	-1.53 (27.1)	-2.33 (9.8)
low	yor	-1.76	-1.11 (10)	-1.77 (22.6)	-2.41 (7.9)
low	sin	-1.83	-1.4 (2.9)	-1.78 (22.4)	-2.31 (10.4)
low	amh	-1.9	-1.46 (1.4)	-1.95 (19.2)	-2.28 (11)
low	npi	-1.96	-1.43 (2.1)	-2.04 (17.6)	-2.41 (7.9)

qwen2.5-72b-instruct

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	1.18	1.47 (72.1)	1.28 (80.3)	0.79 (84.8)
high	zho	1.17	1.29 (67.9)	1.14 (77.8)	1.07 (91.5)
high	fra	1.04	0.96 (60)	1.23 (79.4)	0.92 (87.8)
high	deu	1.02	0.94 (59.3)	1.2 (78.8)	0.94 (88.4)
high	arb	0.86	0.58 (50.7)	1.11 (77.1)	0.89 (87.2)
high	jpn	0.85	0.82 (56.4)	0.96 (74.3)	0.76 (84.1)
high	por	0.84	0.4 (46.4)	1.18 (78.4)	0.94 (88.4)
mid	ron	0.81	0.52 (49.3)	0.97 (74.4)	0.94 (88.4)
high	spa	0.77	0.14 (40)	1.16 (78.1)	1.02 (90.2)
mid	zsm	0.75	0.4 (46.4)	0.97 (74.4)	0.89 (87.2)
mid	ukr	0.72	0.43 (47.1)	0.99 (74.8)	0.74 (83.5)
mid	ind	0.71	0.05 (37.9)	1.13 (77.4)	0.94 (88.4)
high	ita	0.66	0.02 (37.1)	1.04 (75.9)	0.92 (87.8)
mid	kor	0.66	0.28 (43.6)	1 (75)	0.69 (82.3)
high	hin	0.61	0.49 (48.6)	0.52 (65.9)	0.81 (85.4)
mid	tha	0.6	0.43 (47.1)	0.62 (67.8)	0.76 (84.1)
mid	ell	0.54	0.05 (37.9)	0.82 (71.6)	0.76 (84.1)
mid	ben	0.45	-0.1 (34.3)	0.67 (68.8)	0.79 (84.8)
mid	heb	0.42	0.11 (39.3)	0.73 (69.9)	0.43 (76.2)
mid	lit	0.3	-0.54 (23.6)	0.75 (70.3)	0.69 (82.3)
low	npi	-0.07	-0.19 (32.1)	-0.21 (52.1)	0.21 (70.7)
low	kir	-0.24	-0.57 (22.9)	-0.33 (49.8)	0.18 (70.1)
low	tel	-0.51	-0.72 (19.3)	-0.67 (43.4)	-0.15 (62.2)
low	swh	-0.8	-1.2 (7.9)	-0.83 (40.4)	-0.38 (56.7)
low	sin	-1	-1.31 (5)	-0.75 (42)	-0.94 (43.3)
low	hau	-1.19	-1.25 (6.4)	-1.41 (29.3)	-0.91 (43.9)
low	som	-1.27	-1.31 (5)	-1.44 (28.8)	-1.04 (40.9)
low	ibo	-1.29	-1.22 (7.1)	-1.64 (25)	-1.01 (41.5)
low	yor	-1.4	-1.4 (2.9)	-1.55 (26.8)	-1.24 (36)
low	amh	-1.49	-1.46 (1.4)	-1.45 (28.7)	-1.57 (28)

qwen2.5-7b-instruct

resource	lang	Total Z avg.	E-20Q Z (Acc)	MCQ-C Z (Acc)	CR Z (Acc)
high	eng	0.06	-0.66 (20.7)	0.45 (64.7)	0.38 (75)
high	zho	-0.21	-0.49 (25)	-0.27 (51)	0.13 (68.9)
high	spa	-0.28	-1.08 (10.7)	0.06 (57.2)	0.18 (70.1)
high	fra	-0.29	-0.75 (18.6)	0.06 (57.2)	-0.18 (61.6)
high	deu	-0.35	-0.75 (18.6)	-0.11 (54)	-0.2 (61)
mid	ind	-0.41	-1.11 (10)	-0.02 (55.8)	-0.1 (63.4)
high	ita	-0.55	-1.05 (11.4)	-0.54 (45.9)	-0.05 (64.6)
mid	ukr	-0.59	-1.05 (11.4)	-0.43 (47.9)	-0.3 (58.5)
mid	kor	-0.67	-1.2 (7.9)	-0.54 (45.9)	-0.28 (59.1)
mid	zsm	-0.69	-1.02 (12.1)	-0.59 (44.9)	-0.45 (54.9)
mid	ron	-0.78	-1.31 (5)	-0.79 (41.1)	-0.23 (60.4)
high	arb	-0.89	-1.08 (10.7)	-0.45 (47.6)	-1.14 (38.4)
high	por	-0.91	-1.02 (12.1)	-2.01 (18.1)	0.31 (73.2)
mid	lit	-1.19	-1.37 (3.6)	-1.12 (34.9)	-1.06 (40.2)
mid	ell	-1.2	-1.25 (6.4)	-0.91 (38.9)	-1.44 (31.1)
high	jpn	-1.22	-0.93 (14.3)	-0.8 (41)	-1.93 (19.5)
mid	tha	-1.29	-1.4 (2.9)	-1.49 (28)	-0.99 (42.1)
mid	heb	-1.36	-1.43 (2.1)	-0.87 (39.7)	-1.77 (23.2)
high	hin	-1.45	-1.4 (2.9)	-1.48 (28.1)	-1.47 (30.5)
mid	ben	-1.47	-1.31 (5)	-1.57 (26.3)	-1.52 (29.3)
low	tel	-1.52	-1.4 (2.9)	-1.48 (28.1)	-1.67 (25.6)
low	som	-1.62	-1.4 (2.9)	-1.68 (24.3)	-1.77 (23.2)
low	hau	-1.62	-1.4 (2.9)	-1.66 (24.8)	-1.8 (22.6)
low	swh	-1.63	-1.37 (3.6)	-1.59 (26)	-1.93 (19.5)
low	npi	-1.64	-1.37 (3.6)	-1.6 (25.8)	-1.95 (18.9)
low	kir	-1.65	-1.37 (3.6)	-1.45 (28.7)	-2.13 (14.6)
low	yor	-1.74	-1.43 (2.1)	-1.83 (21.6)	-1.95 (18.9)
low	sin	-1.74	-1.46 (1.4)	-1.7 (24)	-2.05 (16.5)
low	ibo	-1.74	-1.46 (1.4)	-1.6 (25.8)	-2.15 (14)
low	amh	-1.78	-1.46 (1.4)	-1.77 (22.7)	-2.1 (15.2)

Table 18: Results for each task on MUG-Eval across 30 languages, evaluated using llama-3.3-70b-instruct, llama-3.1-8b-instruct, qwen2.5-72b-instruct and qwen2.5-7b-instruct. Accuracy was normalized using Z-scores and averaged across tasks. Languages were then ranked by their averaged Z-score.