MAPS: A Multilingual Benchmark for Agent Performance and Security

Omer Hofman[1*†], Jonathan Brokman[1†], Oren Rachmil[1†], Shamik Bose[1], Vikas Pahuja[1], Toshiya Shimizu[2], Trisha Starostina[3], Kelly Marchisio[3], Seraphina Goldfarb-Tarrant[3], Roman Vainshtein[1]

[1]Fujitsu Research of Europe [2]Fujitsu Limited [3]Cohere

Abstract

Agentic AI systems, which build on Large Language Models (LLMs) and interact with tools and memory, have rapidly advanced in capability and scope. Yet, since LLMs have been shown to struggle in multilingual settings, typically resulting in lower performance and reduced safety, agentic systems risk inheriting these limitations. This raises concerns about the accessibility of such systems, as users interacting in languages other than English may encounter unreliable or security-critical agent behavior. Despite growing interest in evaluating agentic AI and recent initial efforts toward multilingual interaction, existing benchmarks do not yet provide a comprehensive, multi-domain, security-aware evaluation of multilingual agentic systems. To address this gap, we propose MAPS, a multilingual benchmark suite designed to evaluate agentic AI systems across diverse languages and tasks. MAPS builds on four widely used agentic benchmarks — GAIA (real-world tasks), SWE-Bench (code generation), MATH (mathematical reasoning), and the Agent Security Benchmark (security). We translate each dataset into eleven diverse languages, resulting in 805 unique tasks and 9,660 total language-specific instances - enabling a systematic analysis of the Multilingual Effect on AI agents' performance and robustness. Empirically, we observe a degradation in both performance and security when transitioning from English to other languages, with severity varying by task and correlating with the amount of translated input. This work establishes the first standardized evaluation framework for multilingual agentic AI, encouraging future research towards equitable, reliable, and accessible agentic AI. `https://huggingface.co/datasets/Fujitsu-FRE/MAPS`

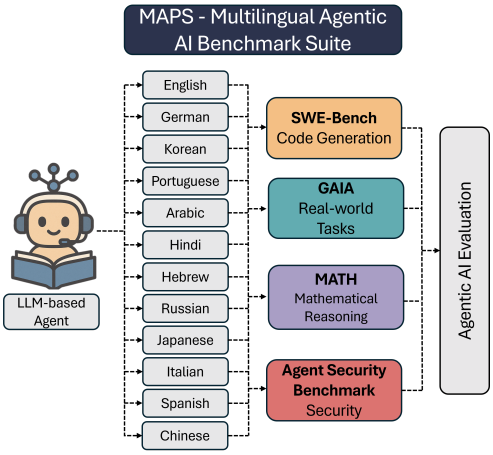


Figure 1: MAPS benchmark suite evaluates LLM-based agents across 12 languages and 4 agentic benchmarks covering performance and security.

1 Introduction

LLM-based agentic AI systems combine multi-step reasoning with external tools and memory to solve open-ended tasks such as code generation, web navigation, planning, and transactional services like booking and ordering (Acharya et al., 2025). By doing so, they extend to complex, real-world problems beyond standard LLMs. As these systems serve speakers of diverse languages, maintaining reliability across languages becomes critical. However, since agentic behavior is grounded in LLMs, which often perform inconsistently across languages (Deng et al., 2023; Wang et al., 2023), agents may inherit these multilingual limitations, reducing their reliability in non-English contexts. While prior multilingual LLM benchmarks have quantified cross-lingual degradation in generation quality(Dang et al., 2024; Shi et al., 2022; Goyal et al., 2022), its impact on *agentic* reasoning, planning, and tool use remains largely unexplored.

In contrast, existing agentic benchmarks span a range of tasks (Mialon et al., 2023; Jimenez et al., 2023; Chang et al., 2024; Xu et al., 2024), but

[*]Corresponding author: omer.hofman@fujitsu.com.
[†]Equal contribution.

Findings of the Association for Computational Linguistics: EACL 2026, pages 821–845
March 24-29, 2026 ©2026 Association for Computational Linguistics

remain English-only. Multilingual evaluations of agentic systems are thus essential for equitable access, as non-English users may face degradation. For instance, a non-English speaker using a banking system with an internal agent may encounter errors even if the system operates in English—issuing a non-English query alone can raise the risk of incorrect tool actions or unsafe behaviors, leading to real-world harm such as erroneous transactions and security vulnerabilities (Zhang et al., 2024).

To address this gap, we introduce MAPS, a **M**ultilingual **A**gentic AI Benchmark Suite for **P**erformance and **S**ecurity - designed to systematically evaluate how multilingual settings affect agent reliability and safety. MAPS is based on four established agentic benchmarks: GAIA (real-world tasks) (Mialon et al., 2023), SWE-Bench (code generation) (Jimenez et al., 2023), MATH (mathematical reasoning) (Hendrycks et al., 2021), and the Agent Security Benchmark (security) (Zhang et al., 2024). These benchmarks are extended to eleven diverse languages beyond English (Fig. 1). By employing a hybrid machine- and LLM-based translation approach Ki and Carpuat (2024) with native-speaker verification, MAPS comprises 805 tasks, each available in 12 language (English plus 11 translations), totaling 9,660 instances.

To demonstrate the use of MAPS, we applied a leading open-source agent from each of the four original benchmarks to its corresponding multilingual extension. We observed notable declines in both task performance and security when shifting from English to other languages, with the severity of these drops varying by task type and correlating with the proportion of non-English input tokens, suggesting that multilingual performance interventions should target input composition and task sensitivity. Beyond overall degradation, we reveal that multilingual inputs also amplify agentic vulnerabilities in safety-critical tasks, highlighting the need for multilingual risk assessment.

This paper makes three primary contributions:

- We introduce MAPS, a multi-domain, security-aware multilingual benchmark suite for agentic AI, extending four widely used benchmarks into eleven diverse languages for performance and security assessment.

- The efficacy and quality of the proposed benchmark are demonstrated through a large-scale evaluation of leading agents as well as human expert verification.

- We present the first quantifiable analysis and evidence that multilingual settings reveal critical performance, safety, and security gaps in agentic systems, along with actionable recommendations for improving their development.

2 Background and Related Work

2.1 Agentic AI Benchmarks

With the rapid advancement of LLM-based agents, a diverse suite of benchmarks has emerged to assess their autonomy, tool use, planning, and memory integration (Yao et al., 2024; Xu et al., 2024; Yehudai et al., 2025). These suites differ from standard LLM evaluations across three main dimensions: **Evaluation objective**: Performance-oriented benchmarks measure task completion, multi-step reasoning, and correct tool use (e.g., AgentBoard Chang et al. (2024)), whereas security-focused suites probe robustness to adversarial inputs and unsafe behaviors (e.g., AgentHarm Andriushchenko et al. (2024); Brokman et al. (2025); Betser et al. (2026)). **Agentic task scope**: Full-agentic evaluations present only problem statements and expected outcomes, requiring end-to-end planning and execution (e.g., GAIA Mialon et al. (2023)), while semi-agentic frameworks supply scaffolding, such as code templates or mock APIs, to isolate the LLM's reasoning (e.g., AppWorld Trivedi et al. (2024)). **Design characteristics**: Most benchmarks span a limited set of domains, typically including real-world information retrieval and navigation (e.g., AssistantBench Yoran et al. (2024)), code generation (e.g., SWE-Bench Jimenez et al. (2023)), reasoning and planning (e.g., MATH Hendrycks et al. (2021), and security scenarios (e.g., Agent Security Benchmark Zhang et al. (2024)). They enable objective measurement despite agents' open-ended capabilities, by formulating tasks with definitive ground truth, allowing clear determination of success (Jimenez et al., 2023; Mialon et al., 2023). Further comparisons appear in the Appendix. Despite their breadth and diversity, nearly all existing agentic benchmarks remain English-only, leaving multilingual behavior unexplored.

2.2 LLMs Multilingual Limitations

Studies show that pre-trained LLMs struggle with non-English inputs, especially in low-resource or typologically distant languages. For instance, multilingual benchmarks (Hu et al., 2020; Liang et al.,

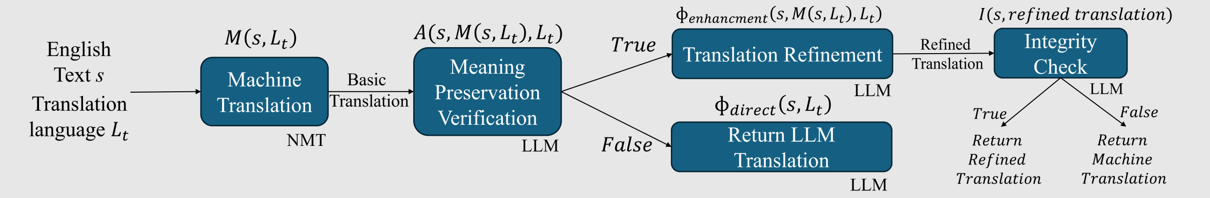


Figure 2: **Overview of our multi-stage translation pipeline for agentic benchmark construction.** We start with machine translation for structural alignment, followed by LLM-based verification and enhancement. This approach is adapted from Ki and Carpuat (2024) but extended with task-specific prompting and fallback mechanisms tailored to the requirements of agentic AI evaluation.

2020) report accuracy drops when moving from English to languages such as Swahili or Nepali. These gaps may stem from imbalanced training data, challenges in tokenizing morphologically rich languages and limited multilingual fine-tuning data (Jha, 2024). Notably, even large models (e.g., GPT-4, Llama 405B) face a "cross-lingual knowledge barrier" (Hendrycks et al., 2020; Chua et al., 2024). LLMs also face security risks in multilingual contexts. Since most security efforts have been English-centric, models are more prone to generate policy-violating outputs when processing non-English prompts (Deng et al., 2023; Wang et al., 2023; Aakanksha et al., 2024). These weaknesses directly propagate to agentic systems, which rely on LLMs for decision-making and tool execution.

2.3 Multilingual LLM Benchmarks

Multilingual LLM evaluation suites such as XTREME (Hu et al., 2020), FLORES (Goyal et al., 2022), and SIB-200 (Adelani et al., 2023) mainly evaluate cross-lingual understanding and generation, covering tasks like translation and cross-lingual retrieval, but do not capture the interactive capabilities that agent benchmarks target.

2.4 Multilingual Agentic Benchmarks

Several recent works have begun to explore multilingual settings in the context of agentic or agent-like systems. For example, X-WebAgentBench (Wang et al., 2025) evaluates multilingual web navigation agents, MASSIVE-Agents (Kulkarni et al., 2025) focuses on multilingual function calling, and WebMMU (Awal et al., 2025) studies multilingual UI understanding and code generation. These efforts reflect a growing recognition of the importance of multilingual interaction in agentic systems.

However, these benchmarks are typically restricted to a single domain or a narrow interaction paradigm (such as web navigation, function calling), and rely on tightly constrained, task-specific agentic scaffolds rather than full, reusable agent frameworks. As a result, they do not capture the diversity of agentic capabilities exercised in realistic deployments, nor do they evaluate security.

MAPS is complementary to these efforts but targets a different point in the design space. Rather than focusing on a single domain, MAPS extends four established agentic benchmarks spanning real-world reasoning, software engineering, mathematical problem solving, and adversarial security scenarios. It evaluates full agent frameworks, with their native tools, and planning loops, under realistic multilingual input conditions.

3 MAPS: Multilingual Agentic AI Benchmark Suite

To support multilingual evaluation of agentic systems, we construct a benchmark suite by extending established English-language datasets into multiple languages. This process requires careful dataset selection, translation procedures that preserve semantic and structural integrity, and mechanisms for ensuring evaluation consistency. The following subsections detail our methodology for translation, benchmark construction, and dataset composition.

3.1 Translation Pipeline

Our translation pipeline combines automated translation with human expert verification - emphasizing both semantic and structural integrity. We combine Neural Machine Translation (NMT), which preserves format and structure (Koehn and Knowles, 2017; Aharoni et al., 2019), and LLM translation for broader contextual fluency. This extends Ki and Carpuat (2024), adding automated quality checks, fallbacks and human expert verification. The overall process is illustrated in Fig. 2. Below is a short summary - further details are in the Appendix.

Automated Translation. This system first computes the NMT translation, then improves it via LLM. let $T : \mathcal{S} \times \mathcal{L} \rightarrow \mathcal{T}$ map an English instruction $s \in \mathcal{S}$ and a target language $L_t \in \mathcal{L}$ to the translation $t \in \mathcal{T}$. We rely on five primitives: $\hat{t} = M(s, L_t)$: neural MT system. $A(s, \hat{t}, L_t) \in \{0, 1\}$: adequacy check of $\hat{t}$. $\Phi_{\text{direct}}(s, L_t)$: direct LLM translation. $\hat{t}_{enh} = \Phi_{\text{enh}}(s, \hat{t}, L_t)$: LLM enhancement of $\hat{t}$. $I(s, \hat{t}) \in \{0, 1\}$: integrity check (hallucination, drift). The translation favors $\hat{t}_{enh}$ with fallbacks as follows:

```
Input: source text $s$, target language $L_t$
Output: translation $t$
$\hat{t} \leftarrow M(s, L_t)$
if $A(s, \hat{t}, L_t) = 0$ then return $\Phi_{\text{direct}}(s, L_t)$   fallback 1
$\hat{t}_{enh} \leftarrow \Phi_{\text{enh}}(s, \hat{t}, L_t)$
if $I(s, \hat{t}_{enh}) = 0$ then return $\hat{t}$                        fallback 2
return $\hat{t}_{enh}$
```

Human Expert Verification. To ensure this pipeline's reliability, we conducted human verification on a representative subset of translations. Evaluation setup and results are in Subsection 3.3.

3.2 Dataset Selection and Composition

Dataset Selection. To enable robust multilingual evaluation of agentic capabilities, we construct MAPS benchmark suite from established agentic AI benchmarks. These were selected based on four criteria: (1) strong research community adoption, including prior use in agentic evaluation; (2) clearly defined, closed-form answers for controlled evaluation; (3) sufficient difficulty to challenge open-source agents; and (4) practical solvability, ensuring that multilingual degradation can be measured. Accordingly, we selected four benchmarks spanning real-world reasoning (GAIA Mialon et al. (2023)), software engineering (SWE-Bench Jimenez et al. (2023)), mathematical problem solving (MATH Hendrycks et al. (2021)), and security assessment (Agent Security Benchmark, ASB Zhang et al. (2024)). Each dataset represents a distinct aspect of agentic behavior-planning, tool use, reasoning, and security. Table 1 summarizes their task types, evaluated capabilities, and metrics, with full dataset details provided in the Appendix.

Data Composition.

Translated Languages. We selected eleven typologically and geographically diverse languages: German, Spanish, Portuguese (Brazil), Japanese, Russian, Chinese, Italian, Arabic, Hebrew, Korean, and Hindi, covering a broad range of scripts, linguistic structures, and regional user populations. Details on the specific dialects translated for each language are provided in the Appendix.

Data Handling. To preserve the integrity of the original benchmarks, we applied only minimal targeted modifications. Translations were appended without altering original metadata (e.g., task type, difficulty level, or tool availability) and domain-specific syntax, such as equations in MATH, code in SWE-Bench, and adversarial prompts in ASB, was preserved exactly. For MATH and SWE-Bench, which were not originally agentic, we retained only the most challenging tasks by difficulty level. This follows common practice in prior work for aligning non-agentic datasets to agentic evaluation (Wu et al., 2023), ensuring meaningful agent evaluation.

Data Volume. To balance performance and security evaluation, MAPS comprises 805 tasks: 405 from performance-oriented datasets (GAIA, SWE-Bench, MATH) and 400 from the ASB. Specifically, we include 165 GAIA tasks (full validation set), 140 high-difficulty MATH problems (20 per topic across 7 topics), and 100 medium-to-hard SWE-Bench tasks, alongside all 400 ASB prompts. Each task was translated into 11 languages and combined with the original English version, yielding 9,660 multilingual instances across 12 languages.

To validate the benchmark's utility and examine the *Multilingual Effects*, we applied a leading agent to each dataset (see Section 4 for details).

Design Rationale: Fixed Environments with Multilingual Inputs. MAPS models realistic deployment settings where users interact in non-English languages while the underlying environment (e.g., documentation, code, or tools) remains primarily English. Translating only the instruction side ensures: (i) *System compatibility:* alignment with current deployments interfacing with English-centric tools; (ii) *Cross-language comparability:* all languages face identical evidence and gold answers, allowing performance and security gaps to be attributed purely to language understanding rather than environmental differences. Agents may use English internally, avoiding artificial constraints. Further discussion appears in Section 7.

3.3 Translation Implementation and Verification

Translation Implementation Details. For neural machine translation, we used Google Translate (Google Cloud, 2024); for LLM translation, Cohere's Command A (Cohere et al., 2025) with a designated system prompt. Structured expres-

Benchmark	Domain / Task Type	Agentic Capability Evaluated	Evaluation Metric
GAIA (Mialon et al., 2023)	Real-world assistant tasks (web, retrieval, reasoning)	Long-horizon planning, autonomous reasoning, and external tool use (browser, code, and document analyzers)	Exact or normalized match to reference answer
SWE-Bench (Jimenez et al., 2023)	Software engineering (bug fixing and validation)	Iterative reasoning with tool feedback, repository navigation, and code-edit planning for patch validation	Test-case success rate (fail-to-pass)
MATH (Hendrycks et al., 2021)	Mathematical problem solving (algebra, geometry, calculus)	Stepwise symbolic reasoning and scratchpad planning; optionally assisted by computational tools (e.g., Wolfram Alpha)	Exact match to final solution
ASB (Zhang et al., 2024)	Security and safety stress testing	Policy adherence, safe tool invocation, and memory integrity under adversarial and injected prompts	Attack success rate / refusal rate

Table 1: Summary of benchmarks included in MAPS. Together, they span diverse domains and evaluation goals, enabling comprehensive multilingual analysis of agentic performance and security.

sions (e.g., code, URLs, equations) were identified via special tokens provided by the datasets (Zhang et al., 2024; Hendrycks et al., 2021), and masked accordingly in the translation process.

Quality Assurance. Translation quality was verified by native-speaking annotators fluent in both English and the target language. Each sample was rated on a 1–5 Likert scale across three established criteria: *adequacy* (semantic fidelity), *fluency* (grammatical and stylistic naturalness), and *formatting accuracy* (preservation of LaTeX, code, etc.) (Freitag et al., 2021). A fourth metric, *answerability*, measured whether the translation preserved intent well enough for the annotator to confidently answer the question as if it were in English. Annotator instructions provided in the Appendix. To validate verifiers' reliability, we embedded *honeypot* samples with intentional errors; annotators reliably flagged these cases, confirming quality control.

Evaluation results confirm high translation quality, with an answerability rate of 94.2%, and mean scores of 4.43 (adequacy), 4.59 (fluency), and 4.75 (formatting) on the 1–5 Likert scale, demonstrating strong semantic fidelity, linguistic naturalness, and structural integrity. A total of 2,200 samples (25% of the benchmark) were manually verified, covering all languages and task types. At this verification scale, statistical analysis shows that even under the most conservative assumptions, the 95% confidence interval for the true corpus-level answerability rate is $[92.0, 96.4]\%$. This suggests that evaluating a quarter of the dataset provides a reasonably precise estimate of overall translation quality, with limited sampling uncertainty. Full per-language results and analysis are included in the Appendix. To support high-precision use cases, we also release a "verified" subset, consisting of approximately 190 translations per language that passed native expert review.

Answerability as key metric. While adequacy and fluency confirm general translation quality, they do not ensure reliable task execution. Our correlation analysis shows that *answerability*, the extent to which a translation remains solvable, most closely aligns with correctness, whereas adequacy and fluency are not predictive. Although this relationship is not absolute since performance naturally also depends on task complexity, answerability remains the most reliable indicator of translation fidelity across languages (see Appendix for details).

4 Experiments

4.1 Experimental Settings

Agent assignment per benchmark. To demonstrate the utility of MAPS, we evaluate one strong, established open-source agent per benchmark, each representing the current state of practice in its respective domain. While a single unified agent would offer a more controlled comparison, such general-purpose agents do not yet exist at sufficient maturity: contemporary agentic systems are tightly specialized around particular toolchains, environments, and prompting strategies, and transferring them across domains typically results in severe performance collapse (Gioacchini et al., 2024; Chang et al., 2024). Importantly, our goal is not to compare agent architectures, but to measure the effect of language on agentic behavior within a fixed agent–environment pair. For each benchmark, the agent, tools, prompts, and LLM backbone and configuration are held constant, and only the input language is varied. This isolates multilinguality as the sole experimental factor, ensuring that any observed degradation reflects language-conditioned

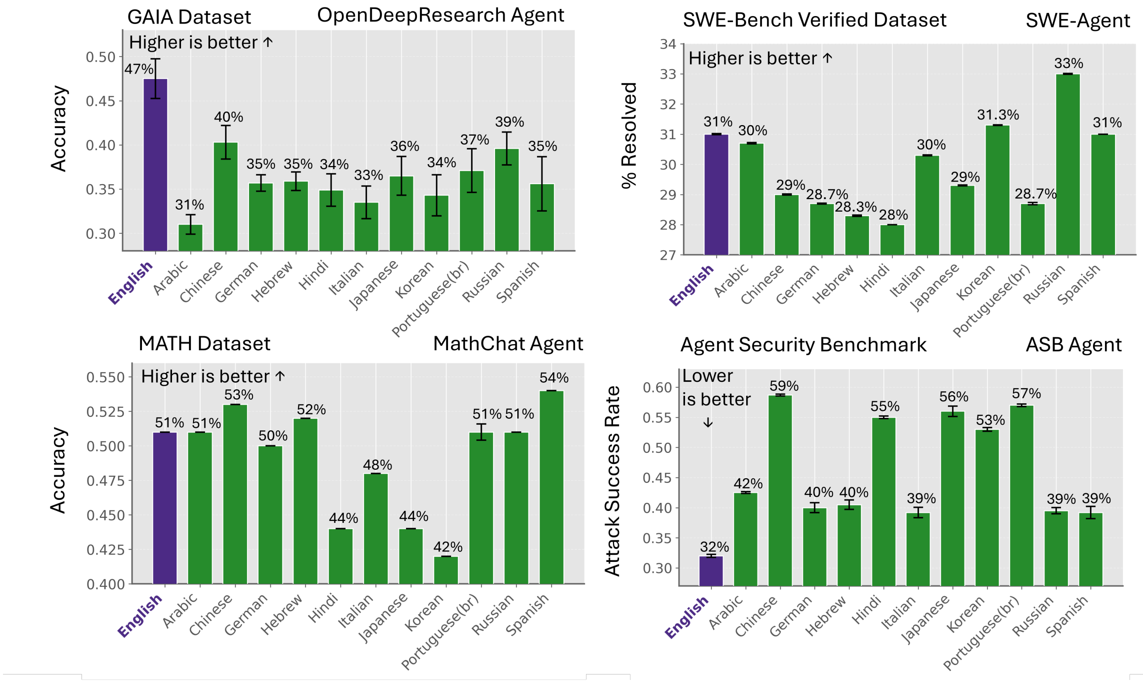


Figure 3: Performance of open-source agents across languages on four agentic benchmarks: GAIA, SWE-Bench, MATH, and ASB. Each bar represents the agent's accuracy (or attack success rate in ASB) for a given language, with English shown as the baseline. Error bars indicate std across three runs. Performance differences reflect each agent's degradation or resilience in multilingual settings.

failure modes rather than architectural differences.

For GAIA, we employ the OpenDeepResearch (ODR) agent (von Platen et al., 2024), integrating retrieval, web browsing, and tool use for real-world reasoning. For MATH, we use MathChat (Wu et al., 2024), a zero-shot multi-turn reasoning agent with Python execution and the Wolfram Alpha tool. For SWE-Bench, we applied SWE-agent (Yang et al., 2024b) for autonomous code reasoning through repository navigation, file editing, and test execution. For ASB, we built on the original benchmark infrastructure (Zhang et al., 2024), eliciting synthetic adversarial tool-use in various agentic scenarios (e.g., travel, legal, system admin). ODR runs with *GPT-o1*, MathChat with *GPT-4*, SWE-Agent with *GPT-4.1*, and ASB with *GPT-5-mini*. Full configuration details are in the Appendix.

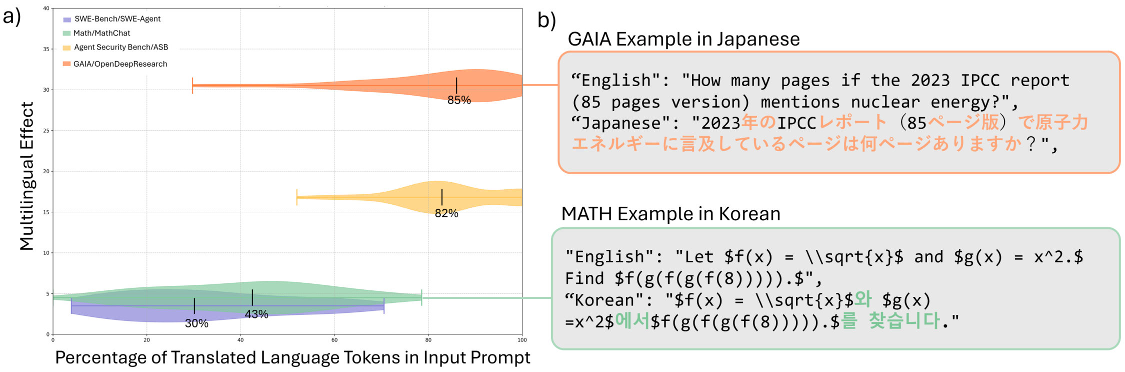
Experiment Protocol. Each agent was evaluated three times per language, across all 12 languages (36 runs per benchmark). We report mean and standard deviation over these runs in Fig. 3.

Metrics. We adopt each benchmark's original evaluation metrics for consistency with prior work: accuracy for MATH and GAIA, test-case success rate for SWE-Bench, and attack success rate (ASR) for ASB. To enable cross-benchmark comparison of multilingual degradation, we further define a unified scalar measure, which we call the Multilingual Effect. For a given benchmark–agent pair, this is computed as the difference between the English performance score and the mean performance score across the 11 non-English languages. For ASB, where higher scores indicate worse security, the sign is inverted so that larger values consistently indicate greater degradation. Intuitively, the Multilingual Effect quantifies how much performance or safety is lost solely due to changing the interaction language. We use this measure throughout the analysis to study how linguistic load, task structure, and agent design modulate multilingual vulnerability.

4.2 Results

Fig. 3 presents agent performance across all four benchmarks in English and the eleven target languages. In GAIA and ASB, we observe a clear Multilingual Effect: non-English languages consistently underperformed compared to English, with reductions of up to 16% (GAIA) and a vulnerability increase of up to 27% in ASB. Notably, SWE-Bench and MATH exhibit only minor variation, with scores clustered around the English baseline.

These results reveal important differences in how multilingual degradation manifests across task types. Although all tasks require complex reason-



```
“English": "How many pages if the 2023 IPCC report
(85 pages version) mentions nuclear energy?",
“Japanese": "2023年のIPCCレポート（85ページ版）で原子力
エネルギーに言及しているページは何ページありますか？",
```

```
"English": "Let $f(x) = \\sqrt{x}$ and $g(x) = x^2.$
Find $f(g(f(g(f(8))))).$",
“Korean": "$f(x) = \\sqrt{x}$와 $g(x)
=x^2$에서$f(g(f(g(f(8))))).$를 찾습니다."
```

Figure 4: a) Multilingual Effect as a function of the proportion of translated language tokens in input prompts. Each point represents a benchmark-agent pair, with the Multilingual Effect computed as the average relative degradation in performance or security across non-English languages. The trend suggests a correlation between input translation extent and multilingual vulnerability. **b)** Two snippets exemplify a low-translation prompt (MATH) and a high-translation prompt (GAIA), clarifying the x-axis percentages in panel (a) and showing how the proportion of natural-language tokens, rather than task difficulty alone, drives the observed Multilingual Effect.

ing, some are more constrained than others. For instance, SWE-Bench and MATH primarily involve structured code or mathematical expressions, thus placing less emphasis on natural language and more on formal syntax or notation, thereby reducing the Multilingual Effect. In contrast, GAIA focuses on solving real-world tasks where understanding flexible natural language prompts is essential. Thus, the importance of the natural language problem statement is significantly higher. To explore this variation, we examine the proportion of localized, target-language text in each benchmark's input (Fig. 4).

Fig. 4 examines the relationship between prompt composition and multilingual performance. Part (a) shows a correlation between the percentage of non-English tokens in the input and the average performance gap (relative to English). Benchmarks with higher proportions of localized, target-language-oriented input, such as GAIA and ASB, exhibit greater degradation, whereas SWE-Bench, with predominantly English input (e.g., code), shows higher preservation. Part (b) contrasts a minimally translated MATH sample example with a nearly fully translated GAIA query

Translation-based ablations. To disentangle translation effects from genuine multilingual degradation, we evaluate two strategies (Table 2). In *deterministic translation*, all non-English prompts are translated to English before inference, isolating translation quality effects. In *agent-decision translation*, the agent decides if and how to translate, reflecting realistic multilingual deployment. Deterministic translation recovers roughly 5% of the lost performance - consistent with our 5.8% human-verified translation error, while the multilingual gap remains higher (GAIA 12%, ASB 14%), indicating that *translation bias* alone cannot explain the degradation. The *agent-decision* variant yields comparable gains, showing partial self-correction through contextual awareness. Yet, both stay below the English baseline, pointing to deeper agent-level reasoning limits across languages.

	English	Mean (Non-English)		
		Orig.	Determin. trans.	Agent-dec. trans.
GAIA (ACC) ↑	47.4%	35.8%	42%	40.9%
ASB (ASR) ↓	32.7%	47.3%	40.4%	43.3%

Table 2: Comparison of deterministic and agent-decision translation on GAIA and ASB. We evaluate two ways to make the agent *aware* of multilingual context: (1) Deterministic translation, where inputs are pre-translated to English, and (2) Agent-decision translation, where the agent decides if and how to translate. Both recover part of the lost performance and robustness, though a clear gap to the English baseline remains.

Cross-Benchmark Evaluation. We examined whether multilingual degradation stems from agent designs or broader task characteristics. Such analyses are constrained by the *benchmark-overfitting problem*, where agents are tightly coupled to domain-specific tools and prompts, limiting cross-task generalization. We identified one viable case: using *OpenDeepResearch* (ODR), our GAIA agent, on the language-light *MATH* benchmark. ODR matched MathChat's accuracy with no multilingual drop, yet degraded by 12% on GAIA, indicating that degradation arises primarily from linguistic

Setting	Benchmark(s)	Configuration	Multilingual Degradation (%)
Cross-Benchmark Check – Same Agent, Different Benchmarks			
ODR	*MATH*	GPT-o1	–0.1%
MATHChat	*MATH*	GPT-4	–0.1%
ODR	*GAIA*	GPT-o1	–12%
Multiple LLM Check – Same Agent, Different Base LLMs			
ODR	*GAIA*	GPT-o1, GPT-4o	–12%,–10%
ASB Agent	*ASB*	GPT-5-mini, Qwen-2	–14.6%,–8.8%

Table 3: Cross-benchmark and cross-model analysis. Results indicate relative degradation compared to the English baseline. Multilingual performance remains stable in language-light tasks (*MATH*), but consistently degrades in language-heavy benchmarks (*GAIA*, *ASB*), independent of the underlying agent or base model.

Benchmark	Original	Unified
GAIA (Acc ↑)	−12%	−10%
ASB (ASR ↓)	+14.6%	+14.6%
MATH (Acc ↑)	−0.1%	−0.9%
SWE-Bench (Acc ↑)	−1.3%	−0.3%

Table 4: Multilingual Effect (relative degradation vs. English) under the original agent setup and under a unified backbone (GPT-5-mini). Language-heavy benchmarks (GAIA, ASB) degrade substantially, while language-light benchmarks (MATH, SWE-Bench) remain stable.

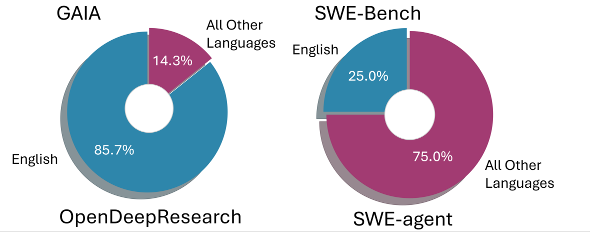


Figure 5: Cross-lingual planning consistency for OpenDeepResearch (ODR) and SWE-Agent. For each agent, we locate tasks solved in English but failed in all other languages, then measure the semantic similarity between the English instruction and each language's initial planning step using multilingual embeddings. ODR exhibits a strong cross-lingual gap: in 85.7% of cases, its English planning step is more faithful to the instruction than in other languages. In contrast, SWE-Agent is more robust, with English leading in only 25% of cases.

load rather than agent architecture (see Table 3).

Cross-Model Evaluation. We next tested whether the multilingual degradation effect is specific to a particular LLM. Replacing *GPT-o1* with *GPT-4o* for ODR and *GPT-5-mini* with *Qwen-2* for ASB yields the same pattern: performance drops in GAIA and elevated ASR in ASB. These consistent trends suggest that multilingual degradation reflects a general property of language-conditioned reasoning, rather than a model-specific bias.

Unified-backbone check. A remaining concern is that multilingual differences across benchmarks might be caused by assigning different base LLMs to different agents (e.g., GPT-o1 for GAIA, GPT-4.1 for SWE-Agent, GPT-5-mini for ASB). While our primary setup preserves each agent's native configuration to avoid unrepresentative failures (Section 4.1), we perform an additional experiment in which all agents are re-evaluated using the same backbone LLM, GPT-5-mini, while keeping all other components (agent logic, prompts, tools, and environments) fixed. Table 4 shows a re-evaluation of all benchmarks under this unified setting, which yields the same qualitative multilingual trends: GAIA and ASB continue to exhibit substantial multilingual degradation, whereas MATH and SWE-Bench remain comparatively stable across languages. Although absolute performance improves due to the stronger base model, the relative multilingual gaps persist, confirming that cross-benchmark differences are driven by task structure rather than model assignment. Full per-language results are provided in the Appendix.

Cross-Lingual Failure Analysis To identify the *root causes* of multilingual degradation, we conduct a deeper analysis of agents' *initial planning behavior* across languages—the first stage where an instruction is decomposed into a sequence of actions. Using multilingual embeddings (OpenAI, 2024b), we measure the cosine similarity between each language's plan and its English counterpart, quantifying how closely non-English reasoning aligns with the original task intent. We focus on tasks solved in English but failed in all other languages, comparing two contrasting agents: *ODR* (GAIA) and *SWE-Agent* (SWE-Bench)—which show the strongest and weakest Multilingual Effects, respectively (Fig. 4).

As shown in Fig. 5, ODR displays a clear English bias: its planning steps align more closely with task instructions in 85% of cases. In contrast, SWE-Agent shows minimal cross-lingual variation, indicating stable reasoning across languages. These results suggest that when an agent's initial planning is language-invariant, multilingual degradation diminishes. A qualitative example of such an error is shown in the appendix.

5 Discussion

5.1 Guidelines for Multilingual Agent Deployment and Risk Assessment

Guidelines for Language-Aware Deployment. Before deploying an AI agent in a multilingual setting, analyze the linguistic composition of its expected input, particularly the balance between structured elements (e.g., code, formal queries) and localized natural language. Inputs with a high proportion of natural language tend to increase the risk of performance and safety degradation. We therefore recommend that, in such a case, developers should conduct a Multilingual Benchmark Assessment using a diverse, language-sensitive evaluation suite, such as MAPS, for AI agents operating across languages. This helps reveal vulnerabilities and promotes reliable global real-world behavior.

Multilingual Inputs Amplify Agentic Security Vulnerabilities. Our ASB evaluation revealed that multilingual adversarial inputs can bypass agent safety mechanisms with minimal sophistication. Direct translations of English jailbreak prompts, without any obfuscation, were sufficient to induce policy-violating behavior across languages, exposing latent vulnerabilities in agent pipelines. Even advanced models such as *GPT-5-mini* proved more resilient in English than in translation. We thus recommend treating multilingual robustness as a *security-critical dimension*, and incorporating translated attacks as a standard component of agentic red-teaming risk assessments.

5.2 Agentic Robustness: Beyond LLM Multilingual Evaluation.

Our results show that multilingual robustness cannot be inferred from LLM-only benchmarks, since agentic systems introduce distinct failure modes absent from text generation. Even when an underlying LLM remains linguistically competent, agents can still fail when language misinterpretations propagate into *operational* errors such as planning drift or tool misuse (Fig. 5), reducing end-to-end reliability. Accordingly, multilingual robustness must be evaluated directly at the *agent level*, where performance and safety reflect language-conditioned decision making rather than text quality alone.

6 Conclusions

We introduce MAPS, the first multilingual benchmark suite for agentic AI systems, addressing a critical gap in assessing language-specific performance and security limitations. By extending four widely used agentic benchmarks—GAIA, SWE-Bench, MATH, and ASB—into eleven diverse languages, our suite enables the analysis of agent behavior under multilingual conditions. Constructed through a hybrid translation pipeline and human verification, MAPS ensures high linguistic and structural consistency. Experiments reveal critical gaps when agents operate in non-English settings, particularly in tasks involving natural language reasoning and safety-critical behavior. These findings underscore the importance of language-aware evaluation and multilingual adaptation, especially for real-world agentic deployments. We view MAPS as a practical foundation for building more inclusive, resilient, and globally reliable agentic AI systems, and invite the community to extend it.

7 Limitations

MAPS introduces the first multilingual suite for evaluating agentic AI systems, spanning four domains and eleven languages. While this represents a strong foundation, expanding into additional domains—such as healthcare, or legal reasoning, and into extremely low-resource languages (e.g., Amharic, Uyghur) would broaden its applicability and strengthen its diagnostic power.

A more specific limitation arises from our decision to vary only the input language, while keeping the agent environments fixed in English. This design reflects the dominant real-world regime where users interact in multiple languages, but the underlying resources—documentation, codebases, schemas, or web content—remain primarily English. While this setup simplifies comparison and isolates linguistic understanding as the primary variable, it does not capture the full complexity of localized ecosystems. Fully non-English environments could introduce additional challenges stemming from tool compatibility, dataset structure, or culturally specific context. Nevertheless, our fixed-environment design offers clear advantages: it ensures cross-language comparability, maintains consistent evidence and outputs, and provides a clean diagnostic for language-conditioned reasoning. We therefore view MAPS as an essential first step toward localized multilingual evaluation—an upper bound on agent robustness that establishes a stable foundation for future extensions.

References

Arash Ahmadian Aakanksha, Beyza Ermis, Seraphina Goldfarb-Tarrant, Julia Kreutzer, Marzieh Fadaee, Sara Hooker, and 1 others. 2024. The multilingual alignment prism: Aligning global and local preferences to reduce harm. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12027–12049.

Deepak Bhaskar Acharya, Karthigeyan Kuppan, and B Divya. 2025. Agentic ai: Autonomous intelligence for complex goals–a comprehensive survey. *IEEE Access*.

David Ifeoluwa Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba O Alabi, Yanke Mao, Haonan Gao, and Annie En-Shiun Lee. 2023. Sib-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. *arXiv preprint arXiv:2309.07445*.

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. *arXiv preprint arXiv:1903.00089*.

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, and 1 others. 2024. Agentharm: A benchmark for measuring harmfulness of llm agents, 2024. *URL https://arxiv.org/abs/2410.09024*.

Rabiul Awal, Mahsa Massoud, Aarash Feizi, Zichao Li, Suyuchen Wang, Christopher Pal, Aishwarya Agrawal, David Vazquez, Siva Reddy, Juan A. Rodriguez, Perouz Taslakian, Spandana Gella, and Sai Rajeswar. 2025. WebMMU: A benchmark for multimodal multilingual website understanding and code generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25118–25145, Suzhou, China. Association for Computational Linguistics.

Roy Betser, Shamik Bose, Amit Giloni, Chiara Picardi, Sindhu Padakandla, and Roman Vainshtein. 2026. Agentrim: Tool risk mitigation for agentic ai. *arXiv preprint arXiv:2601.12449*.

Jonathan Brokman, Omer Hofman, Oren Rachmil, Inderjeet Singh, Vikas Pahuja, Rathina Sabapathy, Aishvariya Priya, Amit Giloni, Roman Vainshtein, and Hisashi Kojima. 2025. Insights and current gaps in open-source llm vulnerability scanners: A comparative analysis. In *2025 IEEE/ACM International Workshop on Responsible AI Engineering (RAIE)*, pages 1–8.

Ma Chang, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. 2024. Agentboard: An analytical evaluation board of multi-turn llm agents. *Advances in Neural Information Processing Systems*, 37:74325–74362.

Zhaorun Chen, Zhen Xiang, Chaowei Xiao, Dawn Song, and Bo Li. 2024. Agentpoison: Red-teaming llm agents via poisoning memory or knowledge bases. *Advances in Neural Information Processing Systems*, 37:130185–130213.

Lynn Chua, Badih Ghazi, Yangsibo Huang, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, Chulin Xie, and Chiyuan Zhang. 2024. Crosslingual capabilities and knowledge barriers in multilingual large language models. *arXiv preprint arXiv:2406.16135*.

Team Cohere, Arash Ahmadian, Marwan Ahmed, Jay Alammar, Milad Alizadeh, Yazeed Alnumay, Sophia Althammer, Arkady Arkhangorodsky, Viraat Aryabumi, Dennis Aumiller, and 1 others. 2025. Command a: An enterprise-ready large language model. *arXiv preprint arXiv:2504.00698*.

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. *Preprint*, arXiv:2412.04261.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. 2024. Agentdojo: A dynamic environment to evaluate attacks and defenses for llm agents. *arXiv preprint arXiv:2406.13352*.

Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. 2023. Multilingual jailbreak challenges in large language models. *arXiv preprint arXiv:2310.06474*.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Luca Gioacchini, Giuseppe Siracusano, Davide Sanvito, Kiril Gashteovski, David Friede, Roberto Bifulco, and Carolin Lawrence. 2024. Agentquest: A modular benchmark framework to measure progress and improve llm agents. *arXiv preprint arXiv:2404.06411*.

Google Cloud. 2024. Translating text using google cloud translation api v3. `https://cloud.google.com/translate/docs/advanced/translating-text-v3#translating_input_strings`. Accessed: July 2025.

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.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

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

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International conference on machine learning*, pages 4411–4421. PMLR.

Basab Jha. 2024. The babel effect: Analyzing multilingual performane discrepancies in large language models. *Eng Appl Sci J*, 1(1):1–4.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. 2023. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*.

Dayeon Ki and Marine Carpuat. 2024. Guiding large language models to post-edit machine translation with error annotations. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4253–4273.

Megan Kinniment, Lucas Jun Koba Sato, Haoxing Du, Brian Goodrich, Max Hasin, Lawrence Chan, Luke Harold Miles, Tao R Lin, Hjalmar Wijk, Joel Burget, and 1 others. 2023. Evaluating language-model agents on realistic autonomous tasks. *arXiv preprint arXiv:2312.11671*.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*.

Mayank Kulkarni, Vittorio Mazzia, Judith Gaspers, Chris Hench, and Jack FitzGerald. 2025. MASSIVE-agents: A benchmark for multilingual function-calling in 52 languages. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 20193–20215, Suzhou, China. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and 1 others. 2020. Xglue: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *arXiv preprint arXiv:2004.01401*.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*.

Tawffeek AS Mohammed. 2025. From google translate to chatgpt: The use of large language models in translating, editing, and revising. In *Role of AI in Translation and Interpretation*, pages 1–32. IGI Global Scientific Publishing.

OpenAI. 2024a. Introducing swe-bench verified. https://openai.com/index/introducing-swe-bench-verified/. Accessed: July 2025.

OpenAI. 2024b. Introducing text-embedding-3-small: a cost-efficient model with improved multilingual and english retrieval performance. https://platform.openai.com/docs/models/text-embedding-3-small. Accessed: July 2025.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Rinna Co. 2025. Lm benchmark. Accessed: 2025-07-28.

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

Yu Tian, Xiao Yang, Jingyuan Zhang, Yinpeng Dong, and Hang Su. 2023. Evil geniuses: Delving into the safety of llm-based agents. *arXiv preprint arXiv:2311.11855*.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. 2024. Appworld: A controllable world of apps and people for benchmarking interactive coding agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16022–16076.

Patrick von Platen, Younes Wang, Teven Le Scao, and Hugging Face team. 2024. Open deep research: A blueprint for transparent foundation model science. https://huggingface.co/blog/open-deep-research. Hugging Face Blog.

Peng Wang, Ruihan Tao, Qiguang Chen, Mengkang Hu, and Libo Qin. 2025. X-WebAgentBench: A multilingual interactive web benchmark for evaluating global agentic system. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19320–19335, Vienna, Austria. Association for Computational Linguistics.

Wenxuan Wang, Zhaopeng Tu, Chang Chen, Youliang Yuan, Jen-tse Huang, Wenxiang Jiao, and Michael R Lyu. 2023. All languages matter: On the multilingual safety of large language models. *arXiv preprint arXiv:2310.00905*.

Yiran Wu, Feiran Jia, Shaokun Zhang, Hangyu Li, Erkang Zhu, Yue Wang, Yin Tat Lee, Richard Peng, Qingyun Wu, and Chi Wang. 2023. Mathchat: Converse to tackle challenging math problems with llm agents. *arXiv preprint arXiv:2306.01337*.

Yiran Wu, Feiran Jia, Shaokun Zhang, Hangyu Li, Erkang Zhu, Yue Wang, Yin Tat Lee, Richard Peng, Qingyun Wu, and Chi Wang. 2024. Mathchat: Converse to tackle challenging math problems with llm agents. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. 2024. Travelplanner: A benchmark for real-world planning with language agents. In *International Conference on Machine Learning*, pages 54590–54613. PMLR.

Frank F Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, and 1 others. 2024. Theagentcompany: benchmarking llm agents on consequential real world tasks. *arXiv preprint arXiv:2412.14161*.

Jianhao Yan, Pingchuan Yan, Yulong Chen, Judy Li, Xianchao Zhu, and Yue Zhang. 2024. Gpt-4 vs. human translators: A comprehensive evaluation of translation quality across languages, domains, and expertise levels. *arXiv preprint arXiv:2407.03658*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024b. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2024. τ-bench: A benchmark for tool-agent-user interaction in real-world domains, 2024. *URL https://arxiv. org/abs/2406.12045*.

Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. 2025. Survey on evaluation of llm-based agents. *arXiv preprint arXiv:2503.16416*.

Jingwei Yi, Yueqi Xie, Bin Zhu, Emre Kiciman, Guangzhong Sun, Xing Xie, and Fangzhao Wu. 2023. Benchmarking and defending against indirect prompt injection attacks on large language models. *arXiv preprint arXiv:2312.14197*.

Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. 2024. Assistantbench: Can web agents solve realistic and time-consuming tasks? *arXiv preprint arXiv:2407.15711*.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. 2024. Injecagent: Benchmarking indirect prompt injections in tool-integrated large language model agents. *arXiv preprint arXiv:2403.02691*.

Hanrong Zhang, Jingyuan Huang, Kai Mei, Yifei Yao, Zhenting Wang, Chenlu Zhan, Hongwei Wang, and Yongfeng Zhang. 2024. Agent security bench (asb): Formalizing and benchmarking attacks and defenses in llm-based agents. *arXiv preprint arXiv:2410.02644*.

A Appendix

Abstract

This supplementary material provides additional details supporting our MAPS benchmark, including extended benchmark comparisons, experimental protocols, and multilingual evaluation results. It is organized as follows:

- **Reproducibility Resources** – links to the full datasets and codebase, to enable open use and distribution for research and development purposes.
- **Agentic AI Benchmark Comparison** – an expanded analysis of existing benchmarks across dimensions such as evaluation objective, task type, answer format, and multilingual support.
- **Hybrid Translation Pipeline** – additional information on our hybrid translation methodology, including domain-specific prompt design and translation configuration experiments.
- **Manual Evaluation of Translation Quality** – annotation guidelines, evaluation examples, and detailed human verification scores per dataset and language.
- **Experimental Settings** – full setup specification, agent configurations, and language adaptation process used to isolate Multilingual Effects across tasks.
- **Extended Results** – detailed multilingual performance tables for GAIA, MATH, SWE-Bench, and ASB across runs.

A.1 Reproducibility Resources

We release the full **MAPS** dataset, which includes translations of four agentic AI benchmarks (GAIA, SWE-Bench, Math, and Agent Security Benchmark) across 12 languages. A verified subset of MAPS, manually reviewed by native speakers, is also included for higher-quality evaluations. Links to both the full dataset and verified subset can be found at: `https://huggingface.co/datasets/Fujitsu-FRE/MAPS`. We also provide the code for our hybrid translation pipeline, including prompt design and format-preserving techniques, available at: `https://github.com/omerhof-fujitsu/hybrid_translation_demo/`.

A.2 Agentic AI Benchmark Comparison

To contextualize MAPS within the growing landscape of agentic evaluation, we provide a detailed comparison of recent benchmarks across core design dimensions. While the main paper (Section 2.1) outlines a conceptual taxonomy—covering evaluation objectives, agent autonomy levels, and domain specialization—this supplementary section complements it with a structured, side-by-side table. Our comparison highlights key properties such as the ground truth availability, domain coverage, and multilingual support. Notably, MAPS remains the only benchmark to offer multilingual assessment across multiple agentic tasks, spanning both performance and safety. This contrast underscores an existing gap in multilingual agent evaluation and illustrates how MAPS fills it.

The columns in Tables 5 and 6 summarize key properties of each benchmark as follows. **Evaluation Objective** specifies what the benchmark is primarily designed to measure. Performance benchmarks evaluate the agent's accuracy to fulfill the given tasks, whereas Security benchmarks probe robustness to adversarial inputs. **Task Scope** describes the degree of freedom given to the agent. Fully agentic benchmarks specify only the task and success criterion, allowing arbitrary agent architectures and strategies, whereas semi-agentic benchmarks provide a partially fixed agentic scaffold (e.g., code templates or tool interfaces) and mainly vary the underlying LLM or prompting. **Domain Coverage** indicates the application domains spanned by the benchmark (e.g., software engineering, mathematics, web navigation, security). **of tasks** reports the number of evaluation instances included in the benchmark. **Answer Format** spec-

Paper	Evaluation Objective	Task Scope	Domain Coverage	# of tasks	Answer Format	GT Available	Multilingual
GAIA (Mialon et al., 2023)	Performance	Full-agentic	General, multimodal	460	Close	✓	✗
SWE Bench (Jimenez et al., 2023)	Performance	Full-agentic	Software engineering	3000	Close	✓	✗
MATH (Hendrycks et al., 2021)	Performance	Full-agentic	Math solver	12,500	Close	✓	✗
τ-bench (Yao et al., 2024)	Performance	Semi-agentic	Airline, retail	165	Close	✓	✗
AgentBoard (Chang et al., 2024)	Performance	Semi-agentic	General – 9 varied tasks	134	Close	✓	✗
App World (Trivedi et al., 2024)	Performance	Semi-agentic	Apps, code execution, APIs	750	Close	Partial	✗
TheAgentCompany (Xu et al., 2024)	Performance	Semi-agentic	SW Dev, HR, Admin, DS	175	Close	✓	✗
ARA (Kinniment et al., 2023)	Performance, Security	Semi-agentic	SW Dev, phishing, web searches	12	Open	✗	✗
TravelPlanner (Xie et al., 2024)	Performance	Full-agentic	Travel planner, tourist	1225	Open	✗	✗

Table 5: Comparison of existing agentic AI benchmarks along evaluation objective, scope, domain coverage, and multilingual availability.

Paper	Evaluation Objective	Task Scope	Domain Coverage	# of tasks	Answer Format	GT Available	Multilingual
Agent Security Bench (Zhang et al., 2024)	Security	Semi-agentic	Offensive cyber operations, unsafe tool	90,000	Close	✓	✗
AgentHarm (Andriushchenko et al., 2024)	Security	Full-agentic	Fraud, cybercrime, harassment	110	Open	✗	✗
AgentDojo (Debenedetti et al., 2024)	Security	Semi-agentic	Prompt injection, phishing, data leakage	97	Both	Partial	✗
InjecAgent (Zhan et al., 2024)	Security	Semi-agentic	Indirect prompt injection, data leakage	1,000	Close	✓	✗
AgentPoison (Chen et al., 2024)	Security	Full-agentic	Backdoor attacks, data poisoning	~20	Open	✗	✗
BIPIA (Yi et al., 2023)	Security	Semi-agentic	Indirect prompt injection, data leakage	5	Close	✓	✗
Evil Geniuses (Tian et al., 2023)	Security	Full-agentic	Prompt injection, data leakage	156	Open	✗	✗

Table 6: Comparison of agentic security benchmarks by scope, threat type, answer format, and multilingual coverage.

ifies how solutions are expressed and evaluated. Closed formats require producing a result in a predefined, structured form with a definitive correct answer, whereas open formats allow free-form outputs for which multiple solutions may be valid. **Ground Truth Available** indicates whether each task has an explicit reference solution. **Multilingual** denotes whether the benchmark natively supports evaluation in multiple languages, as opposed to being designed only for English inputs.

Table 5 surveys general-purpose agentic benchmarks, revealing that most focus exclusively on performance, operate within semi-agentic environments, and lack multilingual coverage. Full-agentic settings—those requiring autonomous planning and execution without scaffolding—are limited to a few benchmarks such as *GAIA*, *SWE-Bench*, and *MATH*, each addressing distinct domains: multimodal reasoning, software engineering, and mathematical problem solving, respectively. Table 6 focuses on security-oriented agentic benchmarks. While these benchmarks advance robustness evaluation, most adopt semi-agentic setups and open-ended answer formats, and often withhold ground truth to prevent overfitting.

Based on this analysis, we selected four complementary benchmarks—*GAIA*, *SWE-Bench*, *MATH*, and *Agent Security Bench (ASB)*—that collectively span performance and security objectives, cover diverse domains including real-world reasoning, code, math, and cybersecurity, and offer closed-form outputs with accessible ground truth. Notably, three of the four operate in fully agentic settings, aligning with our emphasis on end-to-end autonomous decision-making. This configuration enables reliable, multilingual evaluation across both agent capabilities and vulnerabilities, providing a robust foundation for MAPS.

Below, we provide additional details regarding each benchmark:

GAIA. GAIA (Mialon et al., 2023) is a benchmark designed to evaluate agents' performance on real-world assistant tasks. It includes curated questions that require multi-step reasoning and autonomous use of tools such as web browsers, code interpreters, or document analyzers. Each question has a single correct answer, and responses are evaluated by an exact match to a reference output.

SWE-Bench. SWE-Bench (Jimenez et al., 2023) is a software engineering benchmark constructed from real GitHub issues and associated pull requests across popular Python repositories. Each task presents a bug report and a codebase snapshot, and requires the agent to evaluate whether a proposed patch correctly resolves the issue. We adopt the *verified* subset (OpenAI, 2024a), in which agents are tasked with validating a patch rather than generating one.

MATH. The MATH dataset (Hendrycks et al., 2021) includes high-school level mathematical problems across seven topics, including algebra, geometry, and calculus. Tasks are structured to require symbolic manipulation and multi-step reasoning. Agent responses are evaluated by exact match against a reference solution.

Agent Security Benchmark (ASB). ASB benchmark (Zhang et al., 2024) provides a structured evaluation of agent robustness against adversarial threats, including prompt injections, memory poisoning, and tool misuse. Agents interact with injected prompts or environments, and evaluation is based on whether safety policies are violated, measured by attack success rate and refusal rate.

A.3 Hybrid Translation Pipeline Implementation Additional Information

Reliable multilingual evaluation of agentic AI systems hinges on translating task instructions with both semantic and structural cross-language fidelity. Neural MT excels at preserving format and structure, but struggles in low-resource or specialized domains (Koehn and Knowles, 2017; Aharoni et al., 2019). Translation via instructed LLMs offers broader high-level capabilities at the cost of occasional hallucinations and semantic drift (Hendy et al., 2023; Yan et al., 2024). To balance these trade-offs, hybrid pipelines were suggested by Ki and Carpuat (2024); Mohammed (2025), combining format-preserving MT with LLM-based refinement. For MAPS, we extend Ki and Carpuat (2024): First, Ki and Carpuat (2024) was not designed with our benchmarks in mind, thus significant per-benchmark prompting had to be done. Second, we added automated quality checks, fallbacks, and expert verification to ensure the cross-language fidelity needed for agentic benchmarks (Fig. 2).

Formally, let us express our translation pipeline as a function $T : \mathcal{S} \times \mathcal{L} \to \mathcal{T}$, where $s \in \mathcal{S}$ is a task-instruction instance in source-language (English), $L_t \in \mathcal{L}$ is the target language, and $t \in \mathcal{T}$ is the resulting translated output. The pipeline begins with machine translation (MT) to establish a structural foundation: Denote $M(s, L_t)$, the MT function, implemented as a high-quality, off-the-shelf NMT system. Its output provides a structurally faithful baseline for subsequent steps.

Next, we apply a verification step using an LLM to assess whether the translation adequately preserves the source meaning. This is modeled as a binary function $A(s, M(s, L_t), L_t) \to 0, 1$, where the LLM compares the original and translated texts to detect major semantic errors or omissions.

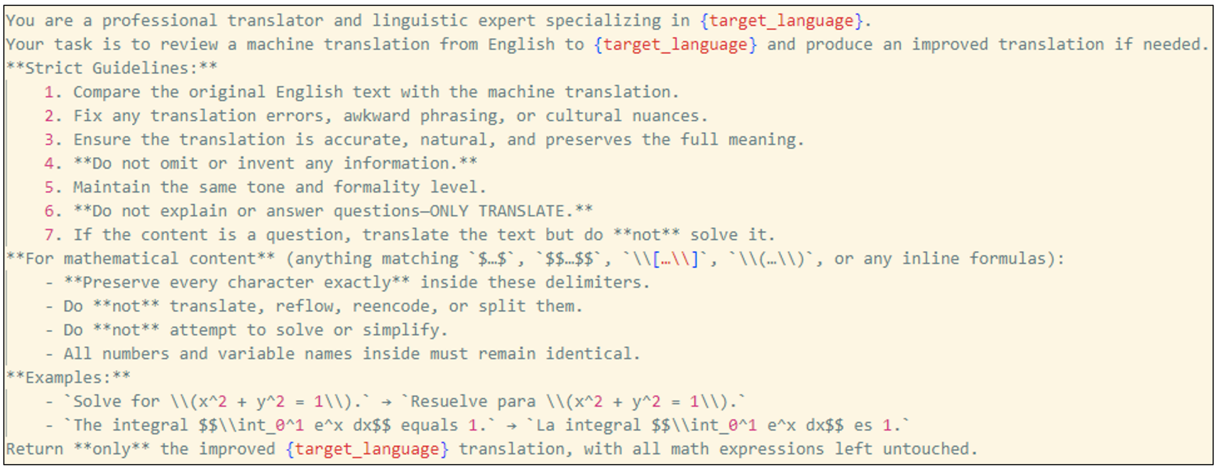
Based on verification outcomes, the pipeline follows one of two distinct paths. If $A = 0$ (indicating MT failure), the pipeline employs direct LLM translation: Denote $\Phi_{\text{direct}}(s, L_t)$ the output of an LLM prompted to directly translate s to language L_t (without using the MT output). If $A = 1$ (indicating acceptable MT), an LLM enhances the translation while preserving its basic structure: Denote $\Phi_{\text{enhancement}}(s, M(s, L_t), L_t)$ as the output of an LLM, guided to improve the MT output while maintaining structural consistency.

To ensure semantic integrity, we apply a second binary check: $I(s, \Phi_{\text{enhancement}}) \to 0, 1$, targeting common LLM failure modes, such as hallucinations, omissions, misinterpretations (e.g., answering instead of translating), and semantic drift. If this verification fails, we revert to the original MT output (which passed the initial verification test).

These conditional steps form a robust decision framework: If MT is rejected, we use a direct LLM translation; if accepted but enhancement fails integrity verification, we fall back to the MT; Otherwise, we use the enhanced translation. Importantly, since both MT and LLM-based translation are robust technologies, we did not encounter any scenario in which both systems failed to produce a valid translation. A formal expression of this pipeline is provided in the Appendix.

For the translation pipeline, we used Python 3.11 alongside a hybrid setup comprising Google's Neural Machine Translation (NMT) API, Cohere Command-A, and GPT-4o. To support high-quality multilingual alignment, we crafted dedicated system prompts for each stage of the pipeline—translation verification, semantic enhancement, and structural integrity checks—tailored to the specific requirements of each domain. For example, GAIA tasks involve real-world decision-making, where preserving subtle intent and contextual detail across languages is critical. In contrast, MATH tasks require precise handling of symbolic notation and mathematical terminology, necessitating strict preservation of LaTeX expressions and domain-appropriate language. An example of a task-specific system prompt is provided in Fig. 6.

To evaluate translation quality across different configurations, we employed LLM-based evaluation using a structured prompt. This approach follows the framework introduced by Kocmi and Federmann (2023), who show that LLMs can reliably assess translation quality when guided by clear criteria. We adapted and extended their methodology to include four dimensions tailored to our



```
You are a professional translator and linguistic expert specializing in {target_language}.
Your task is to review a machine translation from English to {target_language} and produce an improved translation if needed.
**Strict Guidelines:**
    1. Compare the original English text with the machine translation.
    2. Fix any translation errors, awkward phrasing, or cultural nuances.
    3. Ensure the translation is accurate, natural, and preserves the full meaning.
    4. **Do not omit or invent any information.**
    5. Maintain the same tone and formality level.
    6. **Do not explain or answer questions—ONLY TRANSLATE.**
    7. If the content is a question, translate the text but do **not** solve it.
**For mathematical content** (anything matching `$…$`, `$$…$$`, `\\[…\\]`, `\\(…\\)`, or any inline formulas):
    - **Preserve every character exactly** inside these delimiters.
    - Do **not** translate, reflow, reencode, or split them.
    - Do **not** attempt to solve or simplify.
    - All numbers and variable names inside must remain identical.
**Examples:**
    - `Solve for \\(x^2 + y^2 = 1\\).` → `Resuelve para \\(x^2 + y^2 = 1\\).`
    - `The integral $$\\int_0^1 e^x dx$$ equals 1.` → `La integral $$\\int_0^1 e^x dx$$ es 1.`
Return **only** the improved {target_language} translation, with all math expressions left untouched.
```

Figure 6: System prompt used for math translation refinement. This prompt is part of the hybrid translation pipeline and instructs the model to improve machine-translated math questions while preserving all LaTeX expressions and symbolic structure. As part of our hybrid translation pipeline, we designed domain-specific prompts for each benchmark to ensure appropriate evaluation and structural fidelity.

benchmark context (those metrics are similar to the ones we define in the manual verification experiment): (1) adequacy—semantic fidelity to the original task, (2) fluency—naturalness and grammaticality in the target language, (3) formatting accuracy—preservation of structural elements such as LaTeX, code, or tool references, and (4) answerability—whether the translation remains usable and complete. The evaluation prompt used is provided in Fig.7. Unlike traditional evaluation metrics such as COMET (Rei et al., 2020), which rely on the same-language reference translations and embedding-based comparisons, our agentic tasks often lack ground truth references and span cross-language instruction. In such settings, reference-based metrics become inapplicable, whereas LLM-based zero-shot evaluation enables context-sensitive, reference-free scoring.

Using the LLM as a judge approach, we evaluated four translation strategies (Table 7) by translating the GAIA validation set, to empirically identify the most effective pipeline design. This aligns with the analysis in Section 3 of the main manuscript, which discusses the trade-offs between machine translation (MT) and LLM-based generation. As expected, Google NMT showed strong performance on formatting but slightly lagged in semantic adequacy and fluency. Command-A improved fluency and meaning preservation but occasionally introduced structural drift. Hybrid approaches—where NMT outputs were refined by either GPT-4o or Command-A—consistently outperformed single-method setups. Notably, the Google+Command-A variant achieved the highest scores across all metrics, including adequacy (4.76), fluency (4.90), formatting accuracy (4.96), and answerability (95.9%). These results validate our decision to adopt a hybrid approach and extend prior work(Ki and Carpuat, 2024) by integrating domain-specific prompting, automated checks, and structure-aware refinement for multilingual translation in agentic AI benchmarks.

A.4 Hybrid Translation Manual Evaluation Additional Information

To ensure high-fidelity multilingual translation in MAPS, we conducted a human evaluation of translated samples using a standardized annotation protocol. This procedure mirrors the evaluation framework used for our LLM-based assessment, following the methodology proposed by Jiang et al. (Kocmi and Federmann, 2023). Below, we reproduce the instructions provided to annotators, which guided their evaluation of semantic adequacy, linguistic fluency, formatting consistency, and answerability across all benchmark domains.

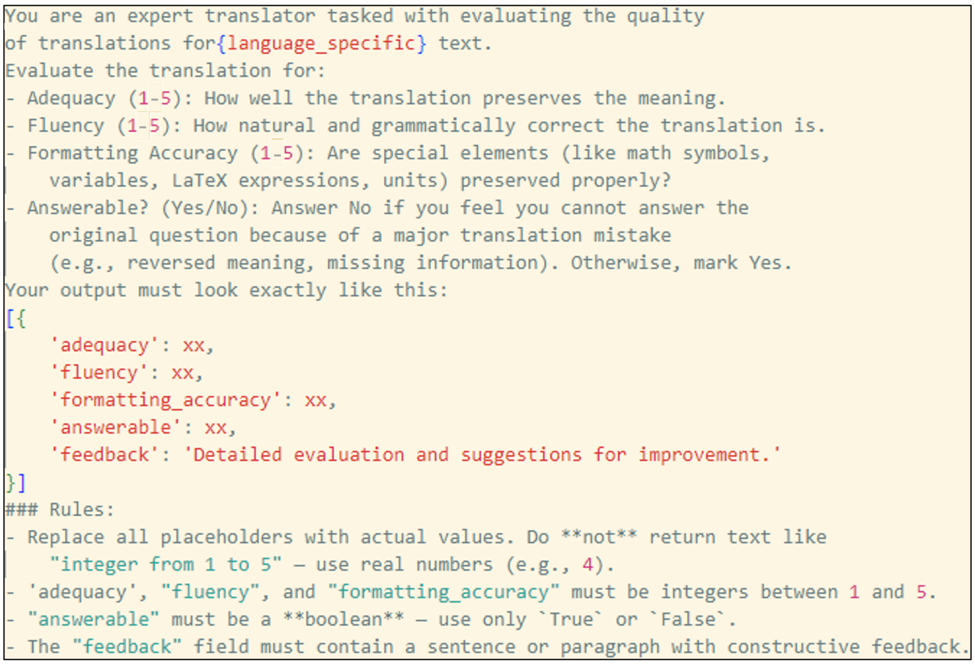
A.4.1 Translation Verification Guidelines for Annotators

Purpose. You are asked to verify the quality of translations from English into your target language for a multilingual benchmark dataset. Your careful evaluation ensures that the translations are accurate, fluent, and correctly formatted.

Dataset Overview and Translation Challenges. You will review 200 translated samples drawn from four benchmark datasets:

Translation Method	Adequacy (1-5)	Fluency (1-5)	Formatting Accuracy (1-5)	Answerable (%)
Google Translate NMT	4.667	4.870	4.946	92.9%
Command A (LLM only)	4.704	4.872	4.934	93.9%
Hybrid (Google Translate NMT + GPT-4o)	4.709	4.832	4.923	94.3%
Hybrid (Google Translate NMT + Command A)	**4.760**	**4.899**	**4.959**	**95.9%**

Table 7: Translation quality comparison across methods based on human evaluation. Bolded values represent the best-performing configuration for each metric.



```
You are an expert translator tasked with evaluating the quality
of translations for{language_specific} text.
Evaluate the translation for:
- Adequacy (1-5): How well the translation preserves the meaning.
- Fluency (1-5): How natural and grammatically correct the translation is.
- Formatting Accuracy (1-5): Are special elements (like math symbols,
    variables, LaTeX expressions, units) preserved properly?
- Answerable? (Yes/No): Answer No if you feel you cannot answer the
    original question because of a major translation mistake
    (e.g., reversed meaning, missing information). Otherwise, mark Yes.
Your output must look exactly like this:
[{
    'adequacy': xx,
    'fluency': xx,
    'formatting_accuracy': xx,
    'answerable': xx,
    'feedback': 'Detailed evaluation and suggestions for improvement.'
}]
### Rules:
- Replace all placeholders with actual values. Do **not** return text like
    "integer from 1 to 5" — use real numbers (e.g., 4).
- 'adequacy', "fluency", and "formatting_accuracy" must be integers between 1 and 5.
- "answerable" must be a **boolean** — use only `True` or `False`.
- The "feedback" field must contain a sentence or paragraph with constructive feedback.
```

Figure 7: System prompt used to guide the LLM-based translation evaluation. The prompt instructs the model to assess translations across four dimensions—adequacy, fluency, formatting accuracy, and answerability—and return results in a strict JSON format.

- **GAIA** (50 samples): natural language questions requiring web search and file analysis.
- **Agent Security Benchmark** (60 samples): tasks probing unsafe or manipulative agent instructions.
- **MATH** (70 samples): algebra, calculus, probability, and geometry problems, often containing LaTeX and variables.
- **SWE-Bench Verified** (20 samples): GitHub issues involving bug descriptions and code snippets.

Evaluation Procedure. For each sample:

1. Read the English source text.
2. Read the translated text in the target language.
3. Assign the following ratings:
    - **Adequacy (1–5):** how well the meaning is preserved.
    - **Fluency (1–5):** how natural and grammatically correct the translation is.
    - **Formatting Accuracy (1–5):** whether LaTeX, variable names, units, and structure are preserved.
    - **Answerable? (Yes/No):** Answer No if you feel you cannot answer the original question because of a major translation mistake (e.g., reversed meaning, missing information). Otherwise, mark Yes.
    - **Notes (optional):** any remarks you wish to add.

Rating Scales.

- **Adequacy:** 5 = perfect meaning preservation, 1 = major meaning loss.
- **Fluency:** 5 = native-like fluency, 1 = broken or ungrammatical.
- **Formatting Accuracy:** 5 = all formatting correct, 1 = major formatting errors.

Important Notes.

- Do not correct or rewrite the translations.
- Pay special attention to LaTeX symbols, variable names, and numerical units.

Sheet Columns Example (one row per sample):

- `Item ID` – Sample identifier.
- `English Source Text` – Original English input.
- `Translated Text` – Translated version.
- `Adequacy (1–5)` – Meaning preservation.
- `Fluency (1–5)` – Naturalness and grammar.
- `Formatting Accuracy (1–5)` – Structural fidelity.
- `Answerable?` – Unanswerable due to error?
- `Notes (Optional)` – Free-text comments.

A.4.2 Translation Verification Extended Results

To ensure the validity of our multilingual benchmark, human verifiers manually reviewed approximately 25% of the translated data—corresponding to over 2,000 samples—across all tasks and languages. In the main manuscript, we reported the overall scores for adequacy, fluency, formatting accuracy, and the error rate derived from answerability. In Tables 10, 11, 12 and 13 we present the full breakdown of results for each individual dataset and in Table 14 we conclude with the manual verification average results across all datasets. Across all four benchmark datasets, native verifiers' ratings reveal high translation quality, with average adequacy (4.47), fluency (4.60), and formatting accuracy (4.76) on a 1–5 scale. Notably, answerability remained consistently high (mean: 0.94), suggesting that most translations retained functional fidelity. Formatting consistency was strongest for ASB and GAIA, while fluency slightly declined for Arabic and Japanese, correlating with minor adequacy drops. Additionally, Fig. 8 presents one successful and one failed translation example, as annotated by Japanese and Korean evaluators, respectively. These results confirm that closed-form multilingual alignment is achievable across diverse tasks without compromising validity.

A.4.3 Statistical sufficiency of the 25% manual audit

Let $n = 2{,}000$ be the size of our manually verified subset ($\approx 25\%$ of the full benchmark). For any binary outcome—here, *Answerable* vs. *Unanswerable*—the sample proportion $\hat{p}$ is an unbiased estimator of the true corpus proportion p. Under the binomial model, its standard error is

$$\mathrm{SE}(\hat{p}) = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}.$$

The variance is maximised when $\hat{p} = 0.5$, giving a worst–case standard error $\mathrm{SE}_{\max} = 0.5/\sqrt{n}$. Using the normal approximation with critical value $z_{0.025} = 1.96$, the 95 % two–sided margin of error is therefore

$$E = 1.96 \times \frac{0.5}{\sqrt{n}} = \frac{0.98}{\sqrt{n}} = \frac{0.98}{\sqrt{2{,}000}} \approx 0.022.$$

Hence any proportion reported from the manual audit has a 95 % confidence interval of at most ± 2.2 percentage points. Put differently, any systematic translation error affecting more than $\approx 2\%$ of the corpus would almost certainly have been detected in our audit. Because the sample was *stratified* by language and dataset, the same formula applies within each stratum with $n_\ell = 0.25\, N_\ell$ items audited per language ℓ. Putting the numbers together, the overall Answerable estimate $\hat{p} = 0.944$ therefore has a 95 % confidence interval

$$0.944 \pm 0.022 \;=\; [0.922,\ 0.966].$$

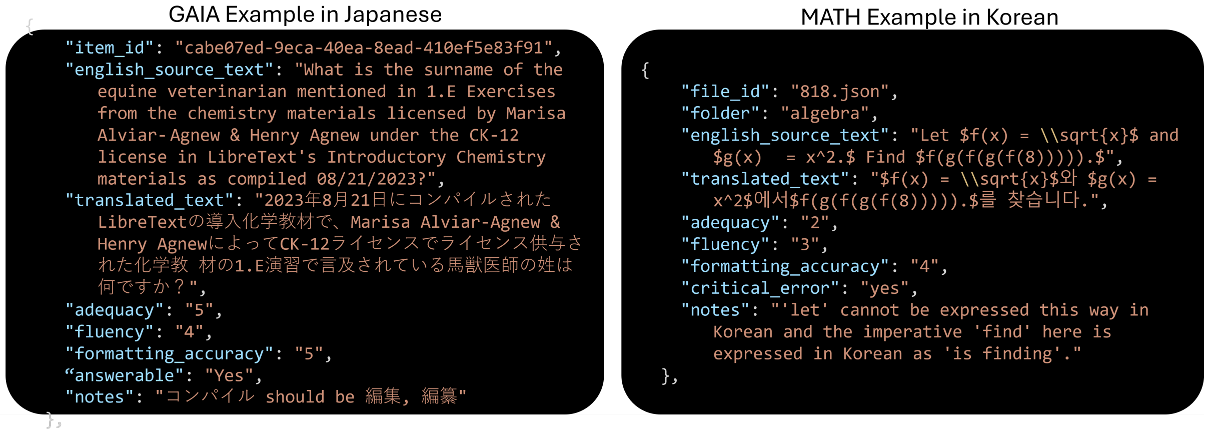
Thus, even in the worst case, the corpus-level answerability remains above 92 %.

Language	Variant
Chinese	Simplified (Mainland)
Russian	Standard (Russia)
Arabic	Modern Standard Arabic
Hebrew	Modern Israeli
German	Standard German
Spanish	Latin American
Hindi	Standard Hindi
Japanese	Standard Japanese
Korean	Standard Korean
Italian	Standard Italian
Portuguese	Brazilian Portuguese

Table 8: Dialect used per language.

A.4.4 Language Variants Used

To ensure wide applicability across regions, we translated the dataset into 11 languages using the



GAIA Example in Japanese

```
{
    "item_id": "cabe07ed-9eca-40ea-8ead-410ef5e83f91",
    "english_source_text": "What is the surname of the
       equine veterinarian mentioned in 1.E Exercises
       from the chemistry materials licensed by Marisa
       Alviar-Agnew & Henry Agnew under the CK-12
       license in LibreText's Introductory Chemistry
       materials as compiled 08/21/2023?",
    "translated_text": "2023年8月21日にコンパイルされた
       LibreTextの導入化学教材で、Marisa Alviar-Agnew &
       Henry AgnewによってCK-12ライセンスでライセンス供与さ
       れた化学教 材の1.E演習で言及されている馬獣医師の姓は
       何ですか？",
    "adequacy": "5",
    "fluency": "4",
    "formatting_accuracy": "5",
    “answerable": "Yes",
    "notes": "コンパイル should be 編集, 編纂"
  },
```

MATH Example in Korean

```
{
    "file_id": "818.json",
    "folder": "algebra",
    "english_source_text": "Let $f(x) = \\sqrt{x}$ and
       $g(x)  = x^2.$ Find $f(g(f(g(f(8))))).$",
    "translated_text": "$f(x) = \\sqrt{x}$와 $g(x) =
       x^2$에서$f(g(f(g(f(8))))).$를 찾습니다.",
    "adequacy": "2",
    "fluency": "3",
    "formatting_accuracy": "4",
    "critical_error": "yes",
    "notes": "'let' cannot be expressed this way in
       Korean and the imperative 'find' here is
       expressed in Korean as 'is finding'."
  },
```

Figure 8: Translation verification examples submitted by annotators during manual evaluation. **Left:** A successful Japanese translation from the GAIA dataset, rated highly across all dimensions with only a minor terminology note. **Right:** A Korean MATH example flagged as critically erroneous due to semantic and grammatical mismatches in imperative phrasing. These examples illustrate the evaluation schema's ability to surface both subtle formatting issues and major adequacy failures.

Feature	Mean AUC	Interpretation
Answerability	0.62	Mild positive relation to accuracy
Formatting	0.55	Very weak positive relation
Adequacy	0.51	≈ 0.5, non-predictive
Fluency	0.49	≈ 0.5, non-predictive

Table 9: Correlation between human-rated translation features and benchmark accuracy across non-English languages. Answerability shows the strongest positive relationship with correctness, while adequacy, fluency, and formatting hover near chance level, indicating limited predictive power.

most widely recognized and formal variant in each case. The table below lists the dialect or regional standard used in the translation process.

A.5 Translation-Quality Correlation Analysis.

To quantify how human-rated translation features relate to task accuracy, we conducted a within-task correlation analysis across all non-English languages. For each task, we computed the area under the ROC curve (AUC) between the binary correctness label ("correct answer") and each translation-quality feature: *adequacy*, *fluency*, *formatting*, and *answerability*. Averaging these per-task AUCs controls for task difficulty and provides a robust measure of how predictive each feature is of successful task transfer.

These results confirm that while adequacy, fluency, and formatting ensure overall translation fidelity, they do not meaningfully predict whether a translated prompt remains solvable. *Answerability* instead captures this functional dimension—whether sufficient information is preserved for the correct reasoning process to still hold—making it the most informative indicator of translation quality for multilingual benchmark reliability.

A.6 Experimental Settings Additional Information

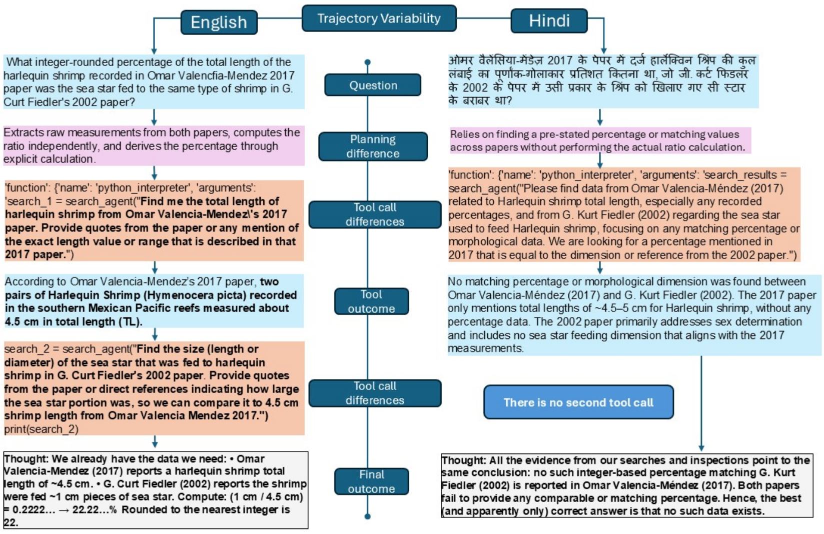
We conducted all experiments on a Linux-based machine (Ubuntu 22.04) equipped with 64 virtual CPUs and 256 GB of RAM. To ensure reproducibility and compatibility, each agent was executed within its own isolated software environment tailored to its specific dependencies. To ensure a fair evaluation of *Multilingual Effects* (see Eq. 2 in the main manuscript), we preserved each agent's original implementation without introducing any code modifications. This isolation strategy allowed us to assess the impact of language variation on agent performance and behavior without confounding factors from architectural or behavioral changes. Datasets were also left unchanged in structure, format, and content. Only the task inputs and expected outputs were translated into the target languages. This process yielded 12 language-specific evaluation suites per dataset (including English), each of which can be executed independently, enabling controlled multilingual evaluation with minimal setup and without requiring code adaptation.

GAIA				
Lang	Adequacy (1–5)	Fluency (1–5)	Formatting Accuracy (1–5)	Answerable
de	4.14	4.04	4.74	0.91
ja	4.30	4.70	4.91	0.88
he	4.20	4.62	4.86	0.88
ko	4.49	4.84	4.40	0.94
pt-br	3.93	4.16	4.53	0.96
it	4.26	4.24	4.72	0.91
es	4.18	4.24	4.99	0.94
ar	3.54	3.79	4.41	0.98
hi	4.22	4.52	4.72	0.92
ru	4.84	4.99	4.90	0.96
AVG	**4.21**	**4.41**	**4.72**	**0.93**

Table 10: Manual translation evaluation scores for the GAIA dataset across 10 languages. Metrics include adequacy, fluency, formatting accuracy (all rated on a 1–5 scale), and answerability (proportion of translations judged as answerable).

MATH				
Lang	Adequacy (1–5)	Fluency (1–5)	Formatting Accuracy (1–5)	Answerable
de	4.80	4.87	4.97	1.00
ja	4.11	4.41	4.96	0.82
he	4.61	4.46	4.79	0.94
ko	4.62	4.95	4.66	0.93
pt-br	4.95	4.84	4.62	1.00
it	4.65	4.85	4.88	1.00
es	4.74	4.49	4.28	0.87
ar	3.78	3.99	3.81	0.82
hi	4.76	4.79	4.96	0.95
ru	4.75	5.00	4.75	0.99
AVG	**4.58**	**4.67**	**4.67**	**0.93**

Table 11: Manual translation evaluation scores for the SWE-Bench dataset across 10 languages.



What integer-rounded percentage of the total length of the harlequin shrimp recorded in Omar Valencfia-Mendez 2017 paper was the sea star fed to the same type of shrimp in G. Curt Fiedler's 2002 paper?

Extracts raw measurements from both papers, computes the ratio independently, and derives the percentage through explicit calculation.

'function': {'name': 'python_interpreter', 'arguments': 'search_1 = search_agent("**Find me the total length of harlequin shrimp from Omar Valencia-Mendez\'s 2017 paper. Provide quotes from the paper or any mention of the exact length value or range that is described in that 2017 paper.**")

According to Omar Valencia-Mendez's 2017 paper, **two pairs of Harlequin Shrimp (Hymenocera picta) recorded in the southern Mexican Pacific reefs measured about 4.5 cm in total length (TL).**

search_2 = search_agent("**Find the size (length or diameter) of the sea star that was fed to harlequin shrimp in G. Curt Fiedler's 2002 paper. Provide quotes from the paper or direct references indicating how large the sea star portion was, so we can compare it to 4.5 cm shrimp length from Omar Valencia Mendez 2017.**")
print(search_2)

Thought: We already have the data we need: • Omar Valencia-Mendez (2017) reports a harlequin shrimp total length of ~4.5 cm. • G. Curt Fiedler (2002) reports the shrimp were fed ~1 cm pieces of sea star. Compute: (1 cm / 4.5 cm) = 0.2222... → 22.22...% Rounded to the nearest integer is 22.

ओमर वैलेंसिया-मेंडेज़ 2017 के पेपर में दर्ज हार्लेक्विन श्रिंप की कुल लंबाई का पूर्णांक-गोलाकार प्रतिशत कितना था, जो जी. कर्ट फिडलर के 2002 के पेपर में उसी प्रकार के श्रिंप को खिलाए गए सी स्टार के बराबर था?

Relies on finding a pre-stated percentage or matching values across papers without performing the actual ratio calculation.

'function': {'name': 'python_interpreter', 'arguments': 'search_results = search_agent("Please find data from Omar Valencia-Méndez (2017) related to Harlequin shrimp total length, especially any recorded percentages, and from G. Kurt Fiedler (2002) regarding the sea star used to feed Harlequin shrimp, focusing on any matching percentage or morphological data. We are looking for a percentage mentioned in 2017 that is equal to the dimension or reference from the 2002 paper.")

No matching percentage or morphological dimension was found between Omar Valencia-Méndez (2017) and G. Kurt Fiedler (2002). The 2017 paper only mentions total lengths of ~4.5–5 cm for Harlequin shrimp, without any percentage data. The 2002 paper primarily addresses sex determination and includes no sea star feeding dimension that aligns with the 2017 measurements.

Thought: All the evidence from our searches and inspections point to the same conclusion: no such integer-based percentage matching G. Kurt Fiedler (2002) is reported in Omar Valencia-Méndez (2017). Both papers fail to provide any comparable or matching percentage. Hence, the best (and apparently only) correct answer is that no such data exists.

Figure 9: Trajectory variability between English and Hindi agents in solving the same question. The English agent follows a computation-driven path—extracting raw values and calculating the result—whereas the Hindi agent relies on surface-level textual matching and assumed equivalence. This leads to reasoning failure due to the absence of explicitly stated percentages in the Hindi query context.

A.6.1 Agents Configuration Additional Information

For the GAIA benchmark, we used the publicly available agent from OpenDeepResearch (von Platen et al., 2024), which features a modular framework supporting document retrieval, browser-based execution, and web search. The agent is designed for general-purpose, real-world reasoning and incorporates memory and planning via tool-

SWE-Bench				
Lang	**Adequacy (1–5)**	**Fluency (1–5)**	**Formatting Accuracy (1–5)**	**Answerable**
de	4.30	4.80	4.50	1.00
ja	4.40	4.05	5.00	0.85
he	4.55	4.50	4.50	0.90
ko	4.35	4.25	4.40	0.90
pt-br	4.75	4.80	4.75	1.00
it	4.55	4.65	4.35	0.95
es	4.00	4.50	4.50	0.95
ar	3.60	4.30	4.20	0.90
hi	4.40	4.35	4.80	0.90
ru	4.85	4.95	4.80	0.95
AVG	**4.375**	**4.515**	**4.580**	**0.930**

Table 12: Manual translation evaluation scores for the SWE-Bench dataset across 10 languages.

ASB				
Lang	Adequacy (1–5)	Fluency (1–5)	Formatting Accuracy (1–5)	Answerable
de	4.10	4.53	4.97	0.95
ja	4.68	4.76	4.91	0.96
he	4.68	4.67	5.00	0.92
ko	4.95	4.93	4.96	0.98
pt-br	4.72	4.55	4.92	1.00
it	4.25	4.47	4.80	0.98
es	4.57	4.40	4.87	0.97
ar	4.45	4.68	4.90	0.98
hi	4.88	4.85	4.95	0.98
ru	4.82	4.90	4.83	0.98
AVG	**4.61**	**4.67**	**4.91**	**0.97**

Table 13: Manual translation evaluation scores for the ASB dataset across 10 languages.

All Benchmarks (Average)				
Lang	Adequacy (1–5)	Fluency (1–5)	Formatting Accuracy (1–5)	Answerable
de	4.375	4.554	4.867	0.964
ja	4.356	4.553	4.936	0.881
he	4.525	4.565	4.840	0.915
ko	4.660	4.847	4.660	0.943
pt-br	4.609	4.580	4.703	0.989
it	4.422	4.565	4.763	0.968
es	4.475	4.403	4.657	0.922
ar	3.859	4.255	4.423	0.936
hi	4.625	4.697	4.881	0.947
ru	4.803	4.961	4.817	0.974
AVG	**4.471**	**4.598**	**4.755**	**0.944**

Table 14: Average manual translation evaluation scores across all benchmarks (GAIA, MATH, SWE-Bench, and ASB). Metrics include adequacy, fluency, and formatting accuracy (rated on a 1–5 scale), and answerability (proportion of samples deemed answerable).

augmented prompting. In our evaluation, we disabled the web search functionality due to its reliance on the external Serper API, which imposes rate limits and introduces latency variability. This omission was applied across all languages to ensure consistency, and throughput scalability under multilingual evaluation. For MATH, we adopted MathChat (Wu et al., 2024), a mathematical reasoning agent that combines multi-turn chain-of-thought prompting with symbolic computation using Python and the Wolfram Alpha API tool.

For SWE-Bench, we used SWE-agent (Yang et al., 2024b), an autonomous software engineering agent capable of navigating repositories, modifying code, and executing test suites. It employs a plan-execute loop enhanced with file-level attention and dynamic tool invocation. For ASB, we built on the original benchmark infrastructure (Zhang et al., 2024), eliciting synthetic adversarial tool-use in various agentic scenarios (e.g. travel-agent, legal consultant, system admin).

Each of the above was executed using the same LLM backbone reported in its original publication. All selected models are considered to be multilingual models. Specifically, GAIA used GPT-O1 (model version: 2024-12-17), MATHChat used

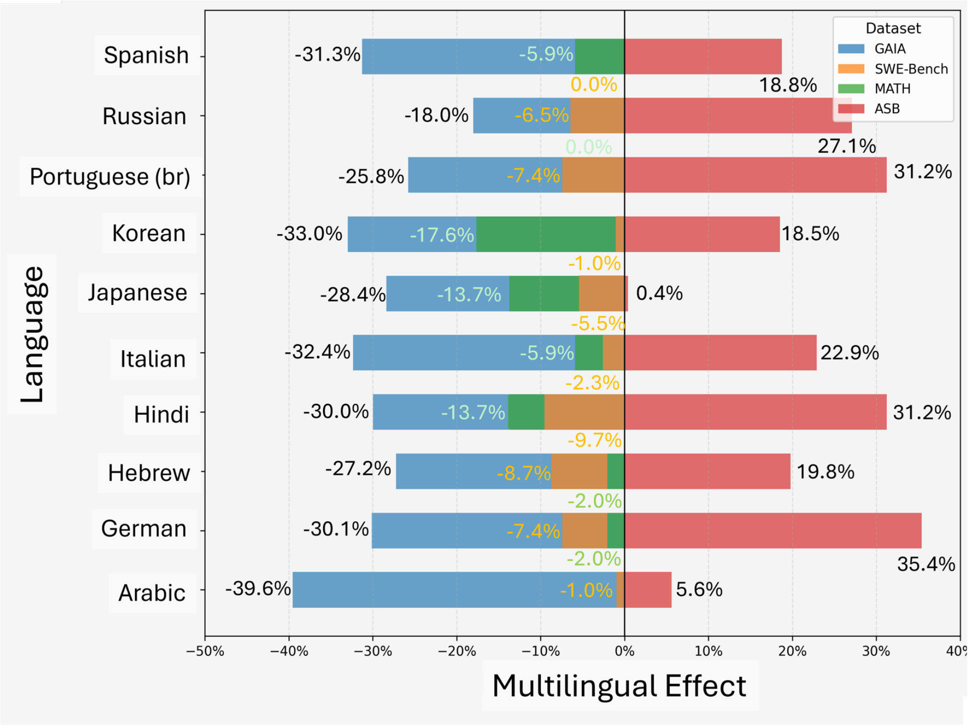


Figure 10: Relative performance differences from English for each language, broken down by dataset. Negative values indicate a drop in performance compared to English, while positive values (notably in ASB) represent increased ASR. The trend highlights how Multilingual Effects vary by language and task type.

GPT-4 (model version: turbo-2024-04-09), SWE-agent used GPT-4.1 (model version: 2025-04-14), and ASB used Qwen2 7B (model version: 2024-10-01).

A.7 Results Additional Information

This section supplements the main manuscript with detailed quantitative and qualitative results.

Cross-Benchmark Evaluation (Extended). A key question in our study is whether the observed multilingual degradation reflects a property of specific agent implementations or a more general limitation of agentic reasoning across languages. To test this, we designed a cross-benchmark experiment—evaluating an agent outside its original domain—to see if the multilingual trend persists.

However, such experiments are inherently constrained by the *benchmark-overfitting problem*: modern agents are typically engineered for narrow domains, with toolchains, prompting strategies, and even reasoning styles optimized for a single benchmark. As noted by Chang et al. (2024); Gioacchini et al. (2024), this coupling makes cross-benchmark evaluation difficult and often misleading, since performance drops may result from tool incompatibility rather than reasoning failure. For example, SWE-Agent's code-compilation tools are irrelevant to GAIA's web-retrieval tasks, and GAIA agents lack the fine-grained debugging environment required for SWE.

Despite these challenges, we identified one feasible cross-benchmark scenario. We evaluated *Open-DeepResearch* (ODR) - our GAIA agent—on the MATH benchmark, where its quantitative reasoning capabilities (via Python/SymPy) remain relevant. ODR was used without modification to its architecture, prompt, or tool configuration. Remarkably, its performance closely matched the benchmark-specific Math-Chat agent (0.1% difference in average accuracy) and exhibited negligible multilingual degradation (0.11%), mirroring Math-Chat's own behavior. In contrast, ODR experiences a 12% multilingual drop on the language-heavy GAIA benchmark.

These findings suggest that multilingual degradation is not tied to a specific implementation or architecture, but rather to the *linguistic load* and reasoning demands of the task environment. In language-light benchmarks such as MATH, both specialized and external agents maintain consistent multilingual performance; in contrast, in linguistically complex benchmarks such as GAIA, ODR performance degrades substantially. Full per-language results are provided in Table 15.

	English	**Arabic**	**Chinese**	**German**	**Hebrew**	**Hindi**	**Italian**	**Japanese**	**Korean**	**Portuguese (BR)**	**Russian**	**Spanish**
MATH-Chat	51.2%	51.8%	53.9%	50.7%	52.9%	44.5%	48.9%	44.7%	42.7%	51.3%	51.8%	54.5%
OpenDeepResearch (ODR)	51.0%	47.2%	48.4%	52.7%	51.5%	47.1%	49.2%	51.0%	48.3%	51.0%	51.2%	48.7%

Table 15: Cross-benchmark evaluation on the **MATH** benchmark. We compare *MATH-Chat*, a benchmark-specific agent, with *OpenDeepResearch (ODR)*, our GAIA agent evaluated under the same multilingual setup. Despite architectural and domain differences, ODR achieves nearly identical performance in English (0.1% difference) and exhibits negligible multilingual degradation (0.11%), supporting the conclusion that cross-lingual degradation depends on linguistic load rather than agent design.

	English	**Arabic**	**Chinese**	**German**	**Hebrew**	**Hindi**	**Italian**	**Japanese**	**Korean**	**Portuguese (BR)**	**Russian**	**Spanish**
GAIA Benchmark (Accuracy ↑, higher is better)												
GPT-O1	47.3%	31.6%	40.8%	35.1%	35.4%	34.0%	33.2%	36.0%	34.5%	37.4%	39.1%	35.0%
GPT-4o	40.1%	29.0%	36.4%	33.9%	29.0%	32.0%	33.3%	31.4%	26.5%	33.9%	32.7%	30.2%
ASB Benchmark (Attack Success Rate ↓, lower is better)												
GPT-5 mini	32.7%	42.5%	58.7%	40.0%	40.5%	55.0%	39.2%	56.0%	53.0%	57.0%	39.5%	39.2%
Qwen 2 7B	48.2%	51.6%	44.7%	65.0%	57.1%	63.3%	59.2%	48.9%	57.0%	63.8%	61.5%	57.6%

Table 16: **Cross-Model Agentic Stability Evaluation.** Comparison of two base LLMs for the same agentic setups across multilingual settings: Qwen 2 7B vs. GPT-5 mini (ASB security benchmark) and GPT-4o vs. GPT-O1 (GAIA performance benchmark). Metrics are reported per language. ASB: Attack Success Rate (↓ lower is better); GAIA: Accuracy (↑ higher is better). Results show that multilingual degradation trends persist independently of the underlying LLM, supporting the findings reported in Section 4.2 in the main paper.

Unified-Backbone Evaluation (extended). Table 19 shows the full per-language results for the unified-backbone experiment described in Section 4.2, in which all four agents are re-evaluated using GPT-5-mini as a shared base LLM, while keeping all other components fixed.

Translation evaluation results (extended). To assess whether multilingual degradation arises primarily from translation quality or from deeper linguistic factors, we evaluated two translation strategies across representative agent–LLM pairs. For each benchmark, we compared: (1) the original multilingual prompts (direct human or NMT translation), (2) the agent-decision mode, in which the agent autonomously decides whether and how to translate its inputs before reasoning or tool invocation, and (3) a deterministic-translation mode, where all non-English inputs are consistently translated into English prior to inference and then back-translated for evaluation. This design isolates the contribution of translation control from broader task-specific differences.

For ODR (GAIA) (Table 17), translation likewise improves performance across most languages, particularly under deterministic control, yet the gap with English persists. Agent-driven translation helps moderately but remains inconsistent, highlighting that model-internal translation heuristics are insufficient for stable reasoning across languages. Overall, these experiments demonstrate that while translation—whether autonomous or deterministic—can partially mitigate multilingual degradation, it cannot eliminate it. A persistent, language-specific effect remains even after all text is nominally brought into English, pointing to deeper representational and alignment disparities in current LLM architectures.

Results for ASB Agent (ASB) are shown in Table 18. Both translation strategies reduce attack-success rates relative to the original multilingual runs, with the deterministic approach showing the clearest improvement for languages such as Italian, Portuguese (BR), and Chinese. However, neither strategy fully restores English-level robustness: even when prompts are deterministically translated, residual cross-lingual variability remains. This suggests that part of the multilingual security gap stems not only from translation noise but also from underlying linguistic and policy-alignment differences within the model itself.

Main Experiment Result (extended). While the main paper summarizes overall trends and Multilingual Effects, Tables 20–23 provide per-language scores, averages, and standard deviations for each benchmark.

Additionally, we have experimented with 2 additional base-models in the GAIA and ASB datasets. Namely, we apply GPT-4o to the OpenDeepResearch agent used in the GAIA dataset and Qwen-2 base model for the ASB agent used in the ASB dataset. Table 16 presents the performance and ASR of both agents with the new base-LLM, re-

	English	**Arabic**	**Chinese**	**German**	**Hebrew**	**Hindi**	**Italian**	**Japanese**	**Korean**	**Portuguese (BR)**	**Russian**	**Spanish**
OpenDeepResearch GPT-O1 (GAIA Benchmark, Accuracy ↑, higher is better)												
Original	47.0%	31.0%	40.0%	35.0%	35.0%	31.0%	33.0%	36.0%	34.0%	37.0%	34.0%	35.0%
Agent decision	–	38.1%	41.8%	39.3%	43.3%	40.0%	37.2%	43.7%	37.6%	41.2%	45.0%	43.1%
Deterministic translation	–	42.0%	42.8%	43.2%	42.6%	38.8%	40.7%	37.6%	39.5%	46.8%	45.6%	42.6%

Table 17: **OpenDeepResearch - ODR (GAIA) Translation Evaluation.** Comparing *Original*, *Agent Decision*, and *Deterministic Translation*. Higher is better (accuracy ↑). All values are percentages.

	English	**Arabic**	**Chinese**	**German**	**Hebrew**	**Hindi**	**Italian**	**Japanese**	**Korean**	**Portuguese (BR)**	**Russian**	**Spanish**
ASB Agent GPT-5 (ASB Benchmark, Attack Success Rate ↓, lower is better)												
Original	32.7%	42.5%	58.7%	40.0%	40.5%	55.0%	39.2%	56.0%	53.0%	57.0%	39.5%	39.2%
Agent decision	–	45.0%	36.0%	38.7%	39.0%	53.0%	40.0%	56.0%	54.0%	39.0%	38.0%	38.0%
Deterministic translation	–	42.5%	35.5%	38.7%	48.0%	48.0%	35.5%	49.0%	44.2%	30.5%	37.2%	35.7%

Table 18: **ASB Agent (ASB) Translation Evaluation.** Comparing *Original*, *Agent Decision*, and *Deterministic Translation*. Lower is better (attack success rate ↓). All values are percentages.

spectively. Interestingly, in the evaluation of the Qwen 7B model in ASB, we observe that Chinese and Japanese yield the lowest ASR in the ASB benchmark, indicating the highest robustness to adversarial prompts in these two Asian languages. We attribute this pattern to the choice of backbone: the ASB agent was implemented using the Qwen2 model (Yang et al., 2024a), which is known for its strong alignment for Chinese and Japanese language tasks. Qwen2 has consistently demonstrated strong performance in Chinese and Japanese-specific LLM benchmarks (Rinna Co., 2025), suggesting that alignment to a particular language and strategic backbone selection can significantly enhance resilience against multilingual adversarial prompts.

Multilingual qualitative failure example. Fig. 9 provides a multilingual qualitative failure example of Hindi language in GAIA, including a comparison of the main steps in English and Hindi.

Language-wise breakdown of performance and ASR results. Fig. 10 provides a language-wise breakdown of relative performance degradation across datasets. From the Figure, we can see that there is no clear correlation between multilingual security robustness (ASB) and multilingual performance degradation. This disconnect is especially clear in real-world, language-heavy tasks like GAIA, where performance drops sharply, while structured tasks like SWE-Bench and MATH remain largely unaffected. This highlights that multilingual security alignment does not directly track with multilingual task accuracy, notably in language-rich agentic tasks.

Benchmark	**EN**	**AR**	**ZH**	**DE**	**HE**	**HI**	**IT**	**JA**	**KO**	**PT-BR**	**RU**	**ES**
ASB (ASR %, ↓ is better)												
ASB	32.7	42.5	58.7	40.0	40.5	55.0	39.2	56.0	53.0	57.0	39.5	39.2
GAIA (Accuracy %, ↑ is better)												
GAIA	55.3	45.5	46.7	48.8	49.7	43.4	46.3	50.9	43.0	46.1	44.2	43.3
MATH (Accuracy %, ↑ is better)												
MATH	82	77	84	83	83	79	83	81	77	81	81	83
SWE-Bench (Accuracy %, ↑ is better)												
SWE-Bench	32	29	31	33	32	30	34	29	35	33	34	29

Table 19: Full per-language results for the unified-backbone experiment, in which all agents are re-evaluated using GPT-5-mini as a shared base LLM. Despite higher absolute performance, the same pattern persists: language-heavy benchmarks (GAIA, ASB) exhibit substantial multilingual degradation, while language-light benchmarks (MATH, SWE-Bench) remain largely stable.

Language	**Run #1**	**Run #2**	**Run #3**	**Mean**	**StdDev**
English (baseline)	0.5094	0.4450	0.4695	0.4746	0.0325
Portuguese (br)	0.3902	0.3658	0.3597	0.3719	0.0161
German	0.3658	0.3475	0.3597	0.3577	0.0093
Spanish	0.3475	0.3536	0.3680	0.3564	0.0105
Italian	0.3536	0.3353	0.3170	0.3353	0.0183
Russian	0.4268	0.4085	0.3530	0.3961	0.0384
Arabic	0.2866	0.3292	0.3170	0.3109	0.0219
Hebrew	0.3818	0.3597	0.3353	0.3589	0.0233
Hindi	0.3353	0.3780	0.3353	0.3496	0.0247
Korean	0.3597	0.3231	0.3475	0.3434	0.0186
Japanese	0.4180	0.3597	0.3170	0.3649	0.0507

Table 20: GAIA benchmark performance across languages.

Language	**Run #1**	**Run #2**	**Run #3**	**Mean**	**StdDev**
English (baseline)	0.29	0.33	0.31	0.31	0.020
Portuguese (br)	0.32	0.29	0.25	0.323	0.0351
German	0.29	0.27	0.30	0.287	0.0153
Spanish	0.31	0.31	0.31	0.31	0.0000
Italian	0.30	0.29	0.32	0.303	0.0150
Russian	0.31	0.34	0.34	0.33	0.0173
Arabic	0.33	0.28	0.31	0.307	0.0252
Hebrew	0.31	0.27	0.27	0.283	0.0231
Hindi	0.28	0.28	0.28	0.280	0.0000
Korean	0.30	0.32	0.32	0.313	0.0115
Japanese	0.31	0.29	0.29	0.297	0.0153

Table 21: SWE-Bench benchmark performance across languages.

Language	**Run #1**	**Run #2**	**Run #3**	**Mean**	**StdDev**
English (baseline)	0.514	0.514	0.514	0.514	0.000
Portuguese (br)	0.514	0.514	0.514	0.514	0.000
German	0.500	0.500	0.500	0.500	0.000
Spanish	0.536	0.536	0.536	0.536	0.000
Italian	0.479	0.479	0.479	0.479	0.000
Russian	0.507	0.507	0.507	0.507	0.000
Arabic	0.514	0.514	0.514	0.514	0.000
Hebrew	0.521	0.521	0.521	0.521	0.000
Hindi	0.436	0.443	0.443	0.440	0.004
Korean	0.414	0.421	0.421	0.419	0.004
Japanese	0.443	0.443	0.443	0.443	0.000

Table 22: MATH benchmark performance across languages.

Language	**Run #1**	**Run #2**	**Run #3**	**Mean**	**StdDev**
English (baseline)	0.534	0.536	0.532	0.534	0.002
Portuguese (br)	0.648	0.657	0.654	0.653	0.005
German	0.659	0.652	0.654	0.655	0.004
Spanish	0.595	0.613	0.597	0.602	0.010
Italian	0.612	0.618	0.603	0.611	0.008
Russian	0.636	0.646	0.642	0.641	0.005
Arabic	0.515	0.504	0.511	0.510	0.005
Hebrew	0.558	0.571	0.572	0.567	0.008
Hindi	0.617	0.616	0.614	0.616	0.002
Korean	0.590	0.589	0.588	0.589	0.001
Japanese	0.484	0.489	0.497	0.490	0.007

Table 23: ASB benchmark performance across languages.