1+1>2: Can Large Language Models Serve as Cross-Lingual Knowledge Aggregators?

Yue Huang[1]*, **Chenrui Fan**[2]*, **Yuan Li**[3], **Siyuan Wu**[4], **Tianyi Zhou**[2], **Xiangliang Zhang**[1]†, **Lichao Sun**[5]

[1]University of Notre Dame [2]University of Maryland, College Park [3]University of Cambridge
[4]Huazhong University of Science and Technology [5]Lehigh University
{yhuang37,xzhang33}@nd.edu cfan42@umd.edu

Abstract

Large Language Models (LLMs) have garnered significant attention due to their remarkable ability to process information across various languages. Despite their capabilities, they exhibit inconsistencies in handling identical queries in different languages, presenting challenges for further advancement. This paper introduces a method to enhance the multilingual performance of LLMs by aggregating knowledge from diverse languages. This approach incorporates a low-resource knowledge detector specific to a language, a language selection process, and mechanisms for answer replacement and integration. Our experiments demonstrate notable performance improvements, particularly in reducing language performance disparity. An ablation study confirms that each component of our method significantly contributes to these enhancements. This research highlights the inherent potential of LLMs to harmonize multilingual capabilities and offers valuable insights for further exploration.

1 Introduction

Large Language Models (LLMs) are increasingly recognized for their impressive capabilities in natural language processing (NLP). Employed across a variety of domains such as the medical sector (Liu et al., 2023c; Zhang et al., 2023a), data generation (Wu et al., 2024), scientific research (Guo et al., 2023; Li et al., 2024c), and LLM-based agents (Liu et al., 2023b; Guo et al., 2024; Huang et al., 2023b; Chen et al., 2024a), LLMs have demonstrated significant utility. Additionally, recent advancements in LLMs have expanded research (Qin et al., 2024; Li et al., 2024a; Xu et al., 2024b; Chen et al., 2024b), which focuses on enhancing their ability to process multiple languages and thereby increasing their accessibility and relevance across diverse linguistic demographics.

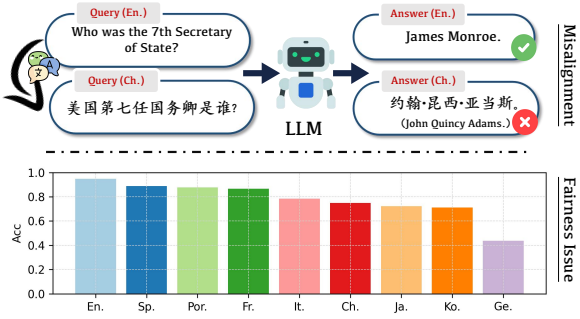

Figure 1: The top is an example of distinct answers to the same questions in different languages. The bottom is the GPT-4's performance on 300 queries in HalluEval (Li et al., 2023a) of nine different languages.

Despite these advancements, LLMs demonstrate inconsistencies when processing queries in different languages with the same meaning (Li et al., 2024d), as evidenced by the results in Figure 1. This inconsistency not only diminishes the efficacy and fairness of LLMs but also signals underlying knowledge conflicts (Xu et al., 2024a) that prevent these models from achieving true intelligence (Liu et al., 2023b; Huang et al., 2023b). Furthermore, such inconsistency can erode trust in LLM applications, particularly when users from varied linguistic backgrounds cannot equally benefit from the technology (Li et al., 2023b).

To address the inconsistency problems in LLMs, we propose a novel method by leveraging the intrinsic capabilities of LLMs through integrating knowledge across different languages. Our approach begins with the development of a low-resource knowledge detector. This detector assesses whether a user's query involves knowledge that is underrepresented in the specific language. When the query does not feature low-resource knowledge, it is directly addressed by the LLMs. In contrast, if low-resource knowledge is detected, the LLMs will be required to select the most relevant target language to handle this specific knowledge. Once the target

*Equal contribution.
†Corresponding anthor.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 13394–13412
November 12-16, 2024

language is selected, the query is translated into this language, and the LLMs generate a response based on the translated query. This response either replaces the original answer or is integrated with it. Finally, the response is translated back to the original language of the query and delivered to the user.

We conducted comprehensive experiments using six popular LLMs and five bilingual datasets (specifically, English and Chinese) to evaluate the effectiveness of our proposed method. The experimental results demonstrate that our approach successfully integrates knowledge from different languages, leading to the improvement in overall performance. Importantly, it significantly reduces the performance disparities across languages, thereby addressing the inconsistency issues inherent in LLMs and promoting fairness for downstream applications. Additionally, our ablation study confirms that both the low-resource knowledge detector and the language selection process are crucial to the improvements observed. Overall, our contributions are as follows:

- We posed an important challenge on the inconsistency of LLMs in downstream tasks, and the low-resource knowledge in a specific language can be brought from another language.
- Based on the observation, we propose a method that utilizes the LLMs' internal capability to enhance its performance on datasets in different datasets through a low-resources knowledge detector, language selection process, and answer replacement & integration.
- We conduct extensive experiments on six popular LLMs and five bilingual datasets. The results show that our proposed method effectively enhances the performance of LLMs by integrating knowledge from different languages and reduce the performance gap in different languages.

2 Related Work

2.1 Multilingual LLMs

There has been a surge in research and work on Multilingual Large Language Models (MLLMs) (Qin et al., 2024; Li et al., 2024a; Xu et al., 2024b; Chen et al., 2024b; Etxaniz et al., 2023). For instance, the InternLM, proposed by Team (2023), is a multilingual language model that has demonstrated excellent performance on multiple Chinese benchmarks. Similarly, PolyLM (Wei et al., 2023b) is another LLM trained using curriculum learning, surpassing other open-source models in multilingual tasks. Besides the above multilingual LLMs, the popular LLMs also include the ChatGLM series developed by Du et al. (2022) and Zeng et al. (2022), and Baichuan series Yang et al. (2023). To improve model performance on multilingual tasks, Muennighoff et al. (2023) and Zhang et al. (2023b) focus on utilizing multilingual training data to fine-tune the parameters. Our work also connects broadly to cross-lingual methods at inference time. Liu et al. (2024) pointed out that translation into English enhances performance for some multilingual tasks, while native language prompting more effectively addresses culturally and linguistically specific questions. In addition, Huang et al. (2023a) and Qin et al. (2023) introduced cross-lingual prompting to enhance the multilingual capabilities of large language models. They focus on improving logical reasoning and task performance across diverse languages. Pourkamali and Sharifi (2024) proposed Self-Supervised Prompting (SSP), a novel method for in-context learning in low-resource languages that improves performance by using stages of noisy labeling and selective exemplar use.

In terms of evaluation, Lai et al. (2023) assessed ChatGPT's performance across 37 different languages. CulturaX (Nguyen et al., 2023) is a multilingual dataset containing 6.3 trillion tokens across 167 languages, aimed at promoting the development of multilingual LLMs. Additionally, M3Exam (Zhang et al., 2023c) introduces a dataset derived from real and official human exam questions, designed for evaluating LLMs in a multilingual, multimodal, and multilevel context. BUFFET consolidates 15 varied tasks across 54 languages into a sequence-to-sequence format, offering a standardized set of few-shot examples and instructions (Asai et al., 2023).

2.2 Factuality in LLMs

One way to improve the factuality of LLMs is the utilization of knowledge graphs (KGs)(Sun et al., 2024b). For instance, Abu-Rasheed et al. (2024) uses knowledge graphs to learn explainable recommendations. Yang et al. (2024b) suggests improving LLMs through the development of knowledge graph-enhanced LLMs, which offers a method to boost the factual reasoning capabilities of LLMs. (Sun et al., 2024a) utilizes the LLM as an agent to interact with and navigate through the KGs, identi-

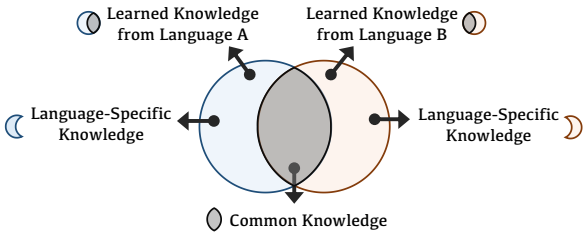

Figure 2: The knowledge domain of a multilingual LLM can be separated into multiple sections (the figure shows two). The language-specific knowledge (pure blue or pure orange) in one language can be utilized for improving the performance in other languages.

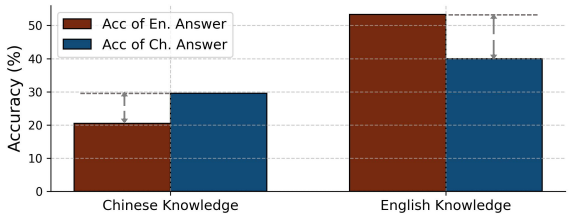

Figure 3: The average performance of six LLMs in five datasets. We show the accuracy of Chinese and English domain knowledge with the query/answer in Chinese and English.

fying relevant entities and relationships, and conducting reasoning with the knowledge it gathers.

Another method to enhance the factual knowledge of LLMs is the utilization of prompt engineering. Previous studies propose various prompt methods such as Chain-of-Thoughts (CoT) (Wei et al., 2023a) and Tree-of-Thoughts (ToT) (Yao et al., 2023). Moreover, some studies use knowledge injection to enhance the domain capability of LLMs (Huang and Sun, 2024; Huang et al., 2024a).

2.3 Hallucination Mitigation

A significant challenge associated with LLMs is their tendency to generate seemingly plausible yet fabricated responses, a phenomenon known as hallucination which is a significant concern in the trustworthiness of LLMs (Huang et al., 2024b, 2023c). To address this issue and prevent misinformation (Huang et al., 2024a), recent research has introduced various hallucination mitigation strategies (Tonmoy et al., 2024). For example, Feng et al. (2024) leverage multi-LLM collaboration to decrease hallucinations in LLM outputs. Additionally, Guan et al. (2024) have developed a novel framework called Knowledge Graph-based Retrofitting (KGR), which integrates LLMs with KGs to minimize factual hallucinations during reasoning. Similarly, Manakul et al. (2023) propose SelfCheckGPT, a sampling method that verifies the accuracy of responses from black-box models without the need for an external database.

3 Methodology

3.1 Motivation

Our proposed method draws inspiration from the distinct knowledge domains inherent to different languages. As illustrated in Figure 2, language-specific knowledge can serve as supplementary information for another language. Figure 3 demonstrates that when queries related to English domain knowledge are posed in Chinese, the performance (i.e., accuracy) of LLMs declines compared to those posed in the English language. Furthermore, Figure 8 reveals that LLMs often provide correct answers in only one of two languages for a given query, suggesting the potential to use the correct response to rectify inaccuracies in the other language. These observations underscore the potential to leverage the strengths of each language to enhance LLM performance across different languages. As shown in Figure 4, the proposed method includes three main modules: *low-resource knowledge detection*, *target language selection*, and *answer replacement & integration*.

3.2 Construction of Low-Resource Dataset

We first construct a low-resource dataset to measure current LLMs' multilingual knowledge transfer capacity. We also use this dataset to train our Low-Resource Knowledge Detector in section 3.3. We initial the dataset with the combination of various existing question-answering datasets including TriviaQA (Joshi et al., 2017), CMMLU (Li et al., 2024b), HalluEval (Li et al., 2023a), TruthfulQA (Lin et al., 2022), MKQA (Longpre et al., 2021), XQuAD (Artetxe et al., 2019), LC-QuAD (Trivedi et al., 2017), KgCLUE (Xu et al., 2020). Moreover, we also construct a dataset that uses LLM-powered synthesized data to cover more knowledge and topics in the training corpus (We call it MULTIGEN). The details of the constructed dataset are shown in Appendix C.

To label these data items, we first use an LLM-Human collaboration to label the samples as Chinese-specific, English-specific, or common sense. Specifically, to confirm the correctness of the labeling, we infer the GPT-4 twice to label the samples with a temperature of 1.0 to enlarge the

potential uncertainty of its output. We then conduct human inspections of the dataset where the labels are inconsistent in two labeling processes, to confirm the labeling and filter out the samples that are too hard or ambiguous for current LLMs. The statistics of the dataset can be found in Table 1.

3.3 Low-Resource Knowledge Detector

The multilingual misalignment stems from the unbalanced training data as the knowledge with low data resources is less likely to be captured by the language model during the pretraining process. For example, queries about the details of Chinese history are not well answered by the model if asked in English as they appear less frequently in the English pretraining corpus. This phenomenon could be improved by fully utilizing the model's inherent capacity. To implement this process, we first adopt a low-resource knowledge detector to identify these low-resource queries and later borrow knowledge from other languages for help.

We train a classifier for each source language to identify the low-resource query for that language. This classifier separates the query about common sense and language-specified knowledge(e.g. Spanish query about Spanish culture) from the low-resources query(e.g. Spanish query about Turkish geography). Queries of the former class are fed into the normal pipeline of language generation while the latter queries are to be enhanced by the knowledge of other languages through our design of other modules. Given a query x in the original language L_o, the low-resource knowledge detector F_{L_o} works as follow:

$$F_{L_o}(x) = \begin{cases} 1 & , \quad x \text{ is low-resource query of } L_o \\ 0 & , \quad \text{else} \end{cases} \tag{1}$$

We demonstrate in the experiment that a classifier is effective enough to distinguish low-resource queries from others. The construction of the training dataset of F_L can be found in subsection 4.1.

The method is cost-effective as it does not require the translation of all queries to multiple languages considering that low-resource query is only a small part of user queries. The majority of user queries are related to common sense and knowledge specified in that language and do not need to go through the following process.

Algorithm 1 Proposed Method

Require: Query x in original language L_o
Ensure: Final answer a_{final}

Low-Resource Knowledge Detection:
Train classifier F_{L_o} for language L_o
$isLowResource \leftarrow F_{L_o}(x)$
if $isLowResource == 1$ **then**
  Target Language Selection:
  Define prompt P_{sel} for selecting target language
  $L_t \leftarrow \text{LLM}(P_{\text{sel}}(x))$
  $x' \leftarrow \text{Trans}(x, L_t)$
  Answer Generation:
  $a_t \leftarrow \text{LLM}(x')$
  $a_o \leftarrow \text{Trans}(a_t, L_o)$
  Answer Integration:
  Define prompt P_{int} for integrating answers
  $a_{\text{final}} \leftarrow \text{LLM}(P_{\text{int}}(a_t, a_o))$
else
  $a_{\text{final}} \leftarrow \text{LLM}(x)$
end if
return a_{final}

3.4 Target Language Selection

After selecting the low-resource query from the user's input, we later adopt a target language selection module to find the most suitable language for that question(e.g. translating a question in English about Chinese history into Chinese). Answering the query with its most resourceful language would improve output quality in terms of correctness and may offer more useful details to the user. We implement this process by prompting the LLM itself as the selection is model-dependent. Different LLMs may select different target languages due to their pretraining corpus. Given the prompt P_{sel} to help select the target language, the low-resource query x, the procedure of Target Language Selection is defined as follows:

$$x' \leftarrow \text{Trans}\big(x, \text{LLM}(P_{\text{sel}}(x))\big), \tag{2}$$

where translator $\text{Trans}(Q, L_t)$ translates the input Q into target language L_t, and LLM is the large language model that selects the most suitable language for x with prompt P_{sel}.

3.5 Answer Replacement & Integration

After translating the original query x to the query in target language x', we use it to prompt the model for the answer in target language a_t. We simply

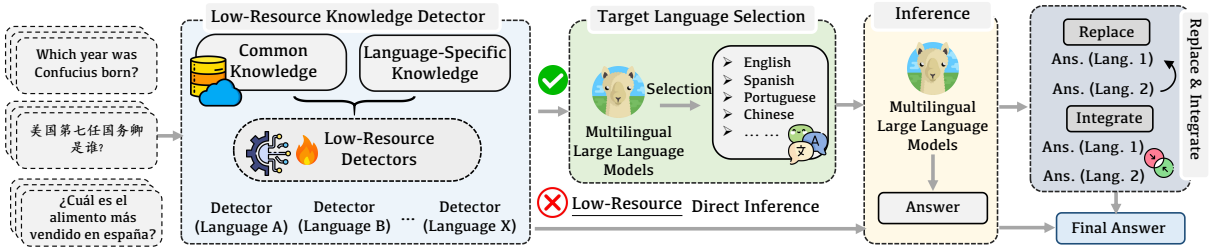

Figure 4: The proposed method begins with the query detection of low-resource knowledge powered by a detector. If low-resource knowledge is detected within the queries, LLMs then select the language most likely to yield the best answer. Answer replacement and integration are employed to formulate the final response.

translate the answer back to the original language to get the final answer a_{final} for the user's understanding.

$$a_{\text{final}} \leftarrow \text{Trans}(a_t, L_o), \tag{3}$$

where L_o is the original language of the user's query.

We also explore the integration of answers in the scenario of open-ended question answering (the prompt template is shown in Appendix F). We let the LLM combine and integrate the answer in the target language a_t and the answer in the original language $a_o = \text{LLM}(x)$:

$$a_{\text{final}} = \text{LLM}(P_{\text{int}}(a_t, a_o)), \tag{4}$$

where P_{int} is the prompt to help LLM integrate between a_t and a_o, and a_{final} is the final answer.

4 Experiments

We chose English and Chinese for our experiments primarily due to their broad applicability and the availability of resources. Firstly, most LLMs, particularly open-source ones like the ChatGLM series, perform significantly in English and Chinese. This trend highlights the advanced development and optimization of LLMs for these languages, making them ideal for rigorous testing. Secondly, major LLM benchmarks and datasets predominantly focus on these two languages. For instance, besides English benchmarks or datasets, benchmarks such as HalluQA and AlignBench are primarily designed around English and Chinese, providing a robust framework for evaluating our methods. Lastly, the linguistic features and data availability in English and Chinese ensure comprehensive evaluation and validation of our approaches and suggest that our findings could be extrapolated to other languages. This potential for cross-linguistic application supports the broader relevance and utility of our study, choosing English and Chinese as both strategic and impactful.

Dataset	Chinese	Common	English	Total	Lang.
TriviaQA	21	754	1040	1815	En.
CMMLU	1200	2162	2751	6113	Ch.
HalluEval	28	923	1033	1984	En.
TruthfulQA	9	322	212	543	En.
MKQA	71	315	1114	1500	En.
XQuAD	72	610	503	1185	En.
LC-QuAD	2	640	345	987	En.
KgCLUE	1218	610	172	2000	Ch.
MULTIGEN	1095	1121	1083	3299	En.
Total	3716	7457	8253	19426	/

Table 1: Dataset statistics of the low-resource knowledge detector. "Lang." is the original language for the dataset.

4.1 Experiment Setup

Training Datasets for Detectors. As we need to train the low-resource detector for each language, for the dataset in English (*e.g.*, TriviaQA) or the dataset in Chinese (*e.g.*, CMMLU, KgCLUE), we translate them to another language (*i.e.*, Chinese or English) through translation API [*].

Detailed Setting. To ensure the reproducibility of results, the temperature parameter for all LLMs is set to 0. For ChatGPT, GPT-4, and Qwen-turbo, we use the official API. For Yi-34b, we use the API from Replicate[†]. For ChatGLM3 and Llama3-Chinese, we deploy them locally for inference with a V100 (40G).

Test Datasets. We selected five datasets for our study, comprising four pre-existing datasets and one that we developed in-house. The following criteria guided our selection:

- The datasets should not predominantly consist of common-sense questions (i.e., questions that

[*] https://fanyi.youdao.com/
[†] https://replicate.com/

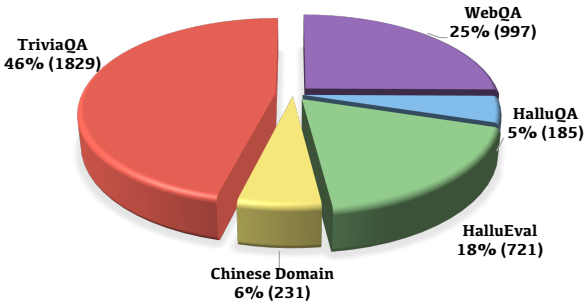

Figure 5: Statistics of the dataset in our experiments.

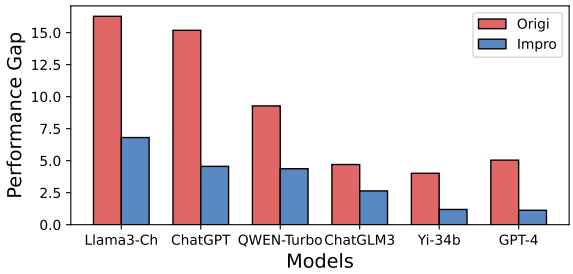

Figure 6: The average performance gap on datasets before and after applying our method.

are independent of linguistic background), as this minimizes the potential for LLMs to demonstrate improvement through linguistic knowledge.

- The datasets should maintain a balance in difficulty; they should not be overly simplistic or excessively challenging. Datasets that are too easy can lead to inflated performance metrics for LLMs, thereby reducing the potential for meaningful improvement. Conversely, datasets that are too challenging can degrade performance across all linguistic contexts, thereby constraining the opportunity to enhance performance in the target language by leveraging knowledge of additional languages.

For all datasets in our study, we select QA-pair samples from them and do not use other extra data to facilitate our evaluation. Totally, we select five datasets for evaluating our method. These include four existing dataset: TriviaQA (Joshi et al., 2017), HaluEval (Li et al., 2023a), HalluQA (Cheng et al., 2023), and WebQA (Li et al., 2016). We show the statistics of the datasets we selected in Figure 5 and the details are shown in Appendix A. In addition to the four datasets mentioned above, we have constructed a bilingual Chinese-English dataset tailored to the Chinese domain. Details of the construction process are provided in Appendix D.

Models. We carefully select six popular LLMs including proprietary and open-source LLMs that master both English and Chinese: ChatGPT (OpenAI, 2023a), GPT-4 (OpenAI, 2023b), ChatGLM3 (Zeng et al., 2022; Du et al., 2022), Yi-34b (AI et al., 2024), Qwen-turbo (Bai et al., 2023), and LLama3-Chinese (lla, 2024).

4.2 Main Results

We evaluate the effectiveness of our proposed method on five benchmark datasets and six popular LLMs mentioned above. Each dataset is translated into a Chinese and an English version for later assessment. We first infer the models with the queries in the dataset to get the generated answers. We then leverage GPT-4 as the judge model to compare each generated answer with the reference answer in the dataset to see if the model produces a correct output. We calculate the generation accuracy and present the result in Table 2. We mark the result in green where there is a significant improvement of more than 1% and mark the result in red if the accuracy decrease by more than 1%.

As can be seen from the table, our method can effectively improve the performance of the model in many scenarios. To be specific, the performance of the GPT-4 model on the HalluEval dataset in Chinese improves significantly from 47.99% to 64.36%. This means there still exists a large cross-lingual knowledge gap in advanced models such as GPT-4 and our method successfully leverages the knowledge across languages to enhance the model's performance. It is important to notice that the improvements do not rely on other models or online resources, they exist due to our leverage of the model's inherent capacity.

It can also be observed from Table 2 that most improvements happen in the language that is different from that of the original dataset, which is also the part where the models suffer from a weaker performance. The comparison of the cross-lingual performance gap before and after applying our method is shown in Figure 6. The figure showcases that our method could significantly reduce the knowledge gap between languages in all LLMs we evaluate, thus improving the fairness of the application for users of different languages.

4.3 Ablation Study

As our generation pipeline consists of several parts, we conduct an ablation study to validate their effectiveness and expenses.

The Impact of the Low-resource Detector. The

Dataset	Lang.	ChatGLM3		ChatGPT		GPT-4		Yi-34b		Qwen-turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
HalluEval	(en)	18.03%	18.03%	57.98%	57.84%	67.13%	67.13%	42.86%	42.72%	29.31%	29.31%	40.67%	40.67%
	(ch)	11.23%	17.34%	32.07%	51.40%	47.99%	64.36%	25.10%	39.67%	19.35%	26.09%	25.35%	37.19%
HalluQA	(en)	20.00%	25.95%	34.27%	30.90%	51.89%	54.05%	38.38%	47.03%	25.97%	37.57%	22.83%	19.57%
	(ch)	22.16%	22.16%	21.91%	24.16%	49.73%	51.35%	45.95%	44.86%	43.65%	43.09%	15.22%	16.30%
Chinese Domain	(en)	9.52%	20.78%	41.85%	42.73%	56.71%	58.44%	33.33%	55.84%	27.19%	46.05%	30.73%	24.24%
	(ch)	32.47%	32.47%	41.85%	41.85%	59.31%	59.74%	63.64%	63.20%	62.28%	61.84%	18.61%	18.61%
triviaQA	(en)	36.32%	36.32%	90.53%	90.37%	94.09%	94.09%	79.33%	79.17%	59.59%	59.47%	77.27%	77.16%
	(ch)	21.33%	31.95%	54.60%	82.67%	82.77%	91.90%	59.43%	75.56%	41.53%	52.99%	43.92%	65.17%
WebQA	(en)	28.51%	38.15%	59.08%	58.88%	67.70%	69.41%	57.07%	68.71%	49.48%	61.08%	50.00%	48.09%
	(ch)	48.69%	48.49%	57.35%	57.86%	72.52%	72.42%	76.93%	76.13%	71.12%	71.33%	37.02%	38.43%

Table 2: Six LLMs' performance on our proposed method.

Dataset	Lang.	Yi-34b		Qwen-turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
HalluEval	(en)	42.86%	41.75%	29.31%	29.59%	40.67%	40.67%
	(ch)	25.10%	39.81%	19.35%	26.51%	25.35%	37.33%
HalluQA	(en)	38.38%	47.03%	25.97%	37.57%	22.83%	18.48%
	(ch)	45.95%	45.95%	43.65%	39.78%	15.22%	20.65%
Chinese Domain	(en)	33.33%	57.58%	27.19%	48.25%	30.74%	24.24%
	(ch)	63.64%	57.14%	62.28%	62.28%	18.61%	22.51%

Table 3: Selected LLMs' performance on the setting without a low-resource detector.

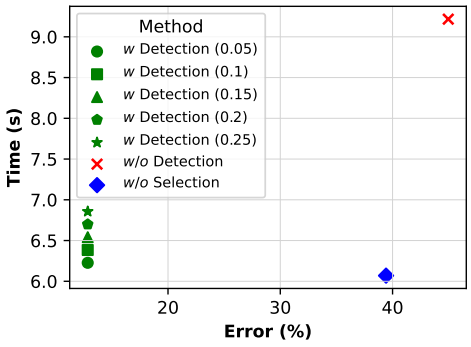

Figure 7: The relationship of time efficiency and error rate. The error rate is the percentage sum of all decreasing in five datasets (value in red on Table 2, Table 3 (*w/o* Detection), Table 4 (*w/o* Selection)).

low-resource detector serves as a filter to sift the language-specific queries from the majority of the queries that involve only commonsense, thus improving efficiency and reducing the expense of the pipeline. As can be observed in Figure 7, a low-resource query detector would significantly reduce the average inference time per sample from more than 9 seconds to less than 6.5 seconds if the ratio of the low-resource queries is 0.05 in the dataset. When the ratio of the low-resource query in the dataset increases, the detector passes more samples into the translation pipeline and increases the average inference time.

Another intriguing finding is that the low-resource detector would increase the model performance. As shown in Table 3, the performance of the pipeline is unstable when we remove the low-resource detector. The overall performance would also drop as we observed in Figure 7. This indicates that the detector and LLM itself can be complementary. The full result of the models' performance without the low-resource detector can be found in Table 7.

The Impact of the Language Selection Module. The language selection module can choose the proper language to answer the question with model-specific choice. It is also flexible in the multi-lingual setting as the resulting target language can be more than two as we test. However, we still validate its effectiveness in the bi-lingual setting, comparing it with the strategy of using the opposite language when the query is detected as low-resources, and show our result in Table 4. The trade-off between its cost and error can also be found in Figure 7.

Dataset	Lang.	Yi-34b		Qwen-turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
HalluEval	(en)	42.86%	42.72%	29.31%	29.17%	40.67%	40.25%
	(ch)	25.10%	42.58%	19.35%	28.61%	25.34%	39.97%
HalluQA	(en)	38.38%	46.49%	25.97%	38.67%	22.83%	21.74%
	(ch)	45.95%	44.86%	43.65%	41.44%	15.21%	15.76%
Chinese Domain	(en)	33.33%	61.04%	27.19%	58.33%	30.74%	19.48%
	(ch)	63.64%	60.17%	62.28%	58.33%	18.61%	18.61%

Table 4: Selected LLMs' performance on the setting without language selection.

As we can see from Figure 7, the language selection module only adds a small inference cost

Type	Lang.	ChatGPT		GPT-4		ChatGLM3		Yi-34b		Qwen-Turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
Integrate	**(ch)**	4.98	5.26	6.90	6.85	4.15	4.09	5.82	5.91	5.88	5.87	3.71	3.58
	(en)	5.92	6.02	7.32	7.54	4.02	4.07	5.86	6.13	5.59	5.78	4.60	4.60
Replace	**(ch)**	4.98	5.47	6.90	6.98	4.15	4.16	5.82	6.23	5.88	6.00	3.71	3.93
	(en)	5.92	5.97	7.32	7.12	4.02	4.26	5.86	6.25	5.59	5.88	4.60	4.54

Table 5: Model performance on AlignBench (Liu et al., 2023a) in the setting of answer replacement and integration.

while significantly improving the model performance. This is due to the existence of the query that is low-resource for both languages, in which switching to the opposite language may make the situation worse. In these situations, the language selection module may pick a third language to better answer the question. The full result of the performance without the language selection module can be found in Table 8.

The Comparison between Answer Replacement and Integration. We further investigated the effectiveness of answer replacement and integration strategies. Given that QA setups with a golden answer may not always accommodate answer integration effectively (for example, when the answers in two different languages factually conflict), we opted for a subset in AlignBench (Liu et al., 2023a) as our evaluation dataset. AlignBench provides a comprehensive, multi-dimensional benchmark designed to assess the alignment of LLMs in Chinese, featuring a variety of open-ended questions. To create a bilingual dataset, we translated the Chinese questions into English. For each response evaluation, we employed an LLM-as-a-judge approach, utilizing the prompt template from AlignBench. The LLM judge then assigned an overall score ranging from 1 to 10 to each LLM response. As indicated in Table 5, both replacement and integration methods significantly enhanced the LLMs' performance across most datasets. Direct replacement led to more substantial improvements but also introduced a higher rate of errors, as evidenced by the performance dips in GPT-4 and Llama3-Ch. Interestingly, the integration method showed a more pronounced performance improvement in English responses, suggesting that LLMs may possess stronger capabilities for answer optimization in English than in Chinese (Yang et al., 2024a).

The Impact of Different Detection Models. As we build a different low-resource detector for each language, the selection of the tokenizer and classification model would impact the training of the detector thereby influencing the overall performance. We adopt language-specific Bert and multi-lingual Bert models to train our low-resource query detector and report the result in Table 6. As shown in the model, using the language-specific model and tokenizer would slightly improve the result of using a multi-lingual model.

Model	Acc.	Recall	Precision	F1.
bert-base-chinese (ch)	86.64	86.64	86.68	86.66
bert-uncased (en)	94.98	94.98	94.88	94.91
Multilingual Bert (ch)	86.47	86.47	86.58	86.51
Multilingual Bert (en)	94.73	94.73	94.64	94.67

Table 6: The impact of model selection on detector training.

5 Discussion on Other Approach

As the confidence of the generated content is related to its entropy during the generation process, a natural idea is to calculate the entropy in different languages and compare them to decide which is the best language to answer the question. This approach is widely used for measuring LLMs' uncertainty and detecting hallucinations (Manakul et al., 2023). However, our trial demonstrates that this approach is infeasible and achieves merely random-guess-level performance when selecting the right language for the given queries.

To explore how to leverage entropy-related statistics to select the target language, we train a model f that takes the statistics as input and outputs the selection of the language Y. The statistics we use for a language l include the entropy of the query E_{Q_l}, the entropy of the response E_{R_l}, the perplexity of the query P_{Q_l}, and the perplexity of the response P_{R_l}. We adopt an MLP as the classification model $f : (E_{Q_l}, E_{R_l}, P_{Q_l}, P_{R_l}) \rightarrow Y$ and train the model on the low-resource query dataset we construct. We trained based on SVM and random forests in Llama2-7b's output. The accuracy is no more than

60%. This is a merely random-guess-level performance when taking the entropy-related statistics as input. We attribute this to the hallucination issue of LLMs, that the model may become overconfident even with the wrong answer (Groot and Valdenegro-Toro, 2024), which indicates LLMs are not calibrated well now (Zhang et al., 2024).

6 Conclusion

This paper presents a method to improve the multilingual capabilities of LLMs by leveraging knowledge from various languages, which includes a low-resource knowledge detector, a process for selecting languages, and answer replacement & integration. Our experiments show significant enhancements in performance, especially in reducing disparities across languages. Moreover, each module in our method contributes to the improvement. Overall, this study underscores the potential of LLMs to unify multilingual functions and provide insights for future research.

Limitations

Our method requires training a separate low-resource query detector for each language. This is not convenient as the developer of a certain language should construct a low-resource training set himself, which involves collecting language-specific data. Also, the dataset should be updated with time with the rise of the new language-specific data.

Ethics Statement

This study adheres to ethical standards in AI research and development. We acknowledge that while our methods aim to enhance the multilingual capabilities of LLMs, they must be implemented with careful consideration of potential biases. Efforts were made to ensure that the knowledge aggregation does not disproportionately favor any particular language or cultural perspective. We also emphasize transparency in our methodologies and findings to enable scrutiny and replication by the research community. The research was conducted without utilizing any personally identifiable information, thereby safeguarding privacy and upholding data protection standards. We commit to ongoing evaluation of our methods in diverse linguistic settings to address and mitigate any emergent biases or disparities. This research seeks not only to advance technology but also to promote inclusivity and fairness in AI applications across different linguistic and cultural groups. In this paper, we utilized AI tools to aid in writing and coding, ensuring that they did not directly contribute to the writing process and that their use adheres to academic standards. Additionally, we ensured that all datasets and benchmarks used in the study comply with their intended purposes and standards.

References

2024. Llama3-chinese. https://github.com/LlamaFamily/Llama-Chinese.

Hasan Abu-Rasheed, Christian Weber, and Madjid Fathi. 2024. Knowledge graphs as context sources for llm-based explanations of learning recommendations.

01. AI, :, Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu, Zhiyuan Liu, and Zonghong Dai. 2024. Yi: Open foundation models by 01.ai.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *CoRR*, abs/1910.11856.

Akari Asai, Sneha Kudugunta, Xinyan Velocity Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov, Sebastian Ruder, and Hannaneh Hajishirzi. 2023. Buffet: Benchmarking large language models for few-shot cross-lingual transfer. *arXiv preprint arXiv:2305.14857*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Liuyi Chen, Yilin Bai, Zhigang He, Chenlong Wang, Huichi Zhou, Yiqiang Li, et al. 2024a. Gui-world: A dataset for gui-oriented multimodal llm-based agents. *arXiv preprint arXiv:2406.10819*.

Du Chen, Yi Huang, Xiaopu Li, Yongqiang Li, Yongqiang Liu, Haihui Pan, Leichao Xu, Dacheng

Zhang, Zhipeng Zhang, and Kun Han. 2024b. Orion-14b: Open-source multilingual large language models.

Qinyuan Cheng, Tianxiang Sun, Wenwei Zhang, Siyin Wang, Xiangyang Liu, Mozhi Zhang, Junliang He, Mianqiu Huang, Zhangyue Yin, Kai Chen, and Xipeng Qiu. 2023. Evaluating hallucinations in chinese large language models.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling.

Julen Etxaniz, Gorka Azkune, Aitor Soroa, Oier Lopez de Lacalle, and Mikel Artetxe. 2023. Do multilingual language models think better in english?

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying llm knowledge gaps via multi-llm collaboration.

Chujie Gao, Qihui Zhang, Dongping Chen, Yue Huang, Siyuan Wu, Zhengyan Fu, Yao Wan, Xiangliang Zhang, and Lichao Sun. 2024. The best of both worlds: Toward an honest and helpful large language model. *arXiv preprint arXiv:2406.00380*.

Tobias Groot and Matias Valdenegro-Toro. 2024. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models.

Xinyan Guan, Yanjiang Liu, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. 2024. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18126–18134.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges.

Taicheng Guo, kehan Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh Chawla, Olaf Wiest, and Xiangliang Zhang. 2023. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. In *Advances in Neural Information Processing Systems*, volume 36, pages 59662–59688. Curran Associates, Inc.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023a. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12365–12394, Singapore. Association for Computational Linguistics.

Yue Huang, Jiawen Shi, Yuan Li, Chenrui Fan, Siyuan Wu, Qihui Zhang, Yixin Liu, Pan Zhou, Yao Wan, Neil Zhenqiang Gong, et al. 2023b. Metatool benchmark for large language models: Deciding whether to use tools and which to use. *arXiv preprint arXiv:2310.03128*.

Yue Huang, Kai Shu, Philip S. Yu, and Lichao Sun. 2024a. From creation to clarification: Chatgpt's journey through the fake news quagmire. In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 513–516, New York, NY, USA. Association for Computing Machinery.

Yue Huang and Lichao Sun. 2024. Fakegpt: Fake news generation, explanation and detection of large language models.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, et al. 2024b. Position: Trustllm: Trustworthiness in large language models. In *International Conference on Machine Learning*, pages 20166–20270. PMLR.

Yue Huang, Qihui Zhang, Lichao Sun, et al. 2023c. Trustgpt: A benchmark for trustworthy and responsible large language models. *arXiv preprint arXiv:2306.11507*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning.

Chong Li, Wen Yang, Jiajun Zhang, Jinliang Lu, Shaonan Wang, and Chengqing Zong. 2024a. X-instruction: Aligning language model in low-resource languages with self-curated cross-lingual instructions.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2024b. Cmmlu: Measuring massive multitask language understanding in chinese.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023a. Halueval: A large-scale hallucination evaluation benchmark for large language models.

Peng Li, Wei Li, Zhengyan He, Xuguang Wang, Ying Cao, Jie Zhou, and Wei Xu. 2016. Dataset and neural recurrent sequence labeling model for open-domain factoid question answering.

Yingji Li, Mengnan Du, Rui Song, Xin Wang, and Ying Wang. 2023b. A survey on fairness in large language models. *arXiv preprint arXiv:2308.10149*.

Yuan Li, Yue Huang, Yuli Lin, Siyuan Wu, Yao Wan, and Lichao Sun. 2024c. I think, therefore i am: Awareness in large language models. *arXiv preprint arXiv:2401.17882*.

Yuan Li, Yue Huang, Hongyi Wang, Xiangliang Zhang, James Zou, and Lichao Sun. 2024d. Quantifying ai psychology: A psychometrics benchmark for large language models. *arXiv preprint arXiv:2406.17675*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods.

Chaoqun Liu, Wenxuan Zhang, Yiran Zhao, Anh Tuan Luu, and Lidong Bing. 2024. Is translation all you need? a study on solving multilingual tasks with large language models. *arXiv preprint arXiv:2403.10258*.

Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Zhuoer Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, et al. 2023a. Alignbench: Benchmarking chinese alignment of large language models. *arXiv preprint arXiv:2311.18743*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. 2023b. Agentbench: Evaluating llms as agents.

Zhengliang Liu, Yue Huang, Xiaowei Yu, Lu Zhang, Zihao Wu, Chao Cao, Haixing Dai, Lin Zhao, Yiwei Li, Peng Shu, et al. 2023c. Deid-gpt: Zero-shot medical text de-identification by gpt-4. *arXiv preprint arXiv:2303.11032*.

Shayne Longpre, Yi Lu, and Joachim Daiber. 2021. Mkqa: A linguistically diverse benchmark for multilingual open domain question answering.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages.

OpenAI. 2023a. Chatgpt. https://openai.com/product/chatgpt.

OpenAI. 2023b. Gpt-4. https://openai.com/gpt-4.

Nooshin Pourkamali and Shler Ebrahim Sharifi. 2024. Machine translation with large language models: Prompt engineering for persian, english, and russian directions. *arXiv preprint arXiv:2401.08429*.

Libo Qin, Qiguang Chen, Fuxuan Wei, Shijue Huang, and Wanxiang Che. 2023. Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2695–2709, Singapore. Association for Computational Linguistics.

Libo Qin, Qiguang Chen, Yuhang Zhou, Zhi Chen, Yinghui Li, Lizi Liao, Min Li, Wanxiang Che, and Philip S. Yu. 2024. Multilingual large language model: A survey of resources, taxonomy and frontiers.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel M. Ni, Heung-Yeung Shum, and Jian Guo. 2024a. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024b. Head-to-tail: How knowledgeable are large language models (llms)? a.k.a. will llms replace knowledge graphs?

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

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. 2017. Lc-quad: A corpus for complex question answering over knowledge graphs. In *International Semantic Web Conference*, pages 210–218. Springer.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023a. Chain-of-thought prompting elicits reasoning in large language models.

Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Bowen Yu, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. 2023b. Polylm: An open source polyglot large language model.

Siyuan Wu, Yue Huang, Chujie Gao, Dongping Chen, Qihui Zhang, Yao Wan, Tianyi Zhou, Xiangliang Zhang, Jianfeng Gao, Chaowei Xiao, et al. 2024. Unigen: A unified framework for textual dataset generation using large language models. *arXiv preprint arXiv:2406.18966*.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. 2020. CLUE: A Chinese language understanding evaluation benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4762–4772, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Rongwu Xu, Zehan Qi, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024a. Knowledge conflicts for llms: A survey.

Yuemei Xu, Ling Hu, Jiayi Zhao, Zihan Qiu, Yuqi Ye, and Hanwen Gu. 2024b. A survey on multilingual large language models: Corpora, alignment, and bias.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2024a. Large language models as optimizers.

Linyao Yang, Hongyang Chen, Zhao Li, Xiao Ding, and Xindong Wu. 2024b. Give us the facts: Enhancing large language models with knowledge graphs for fact-aware language modeling.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2023. Large language model as attributed training data generator: A tale of diversity and bias.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Kai Zhang, Jun Yu, Zhiling Yan, Yixin Liu, Eashan Adhikarla, Sunyang Fu, Xun Chen, Chen Chen, Yuyin Zhou, Xiang Li, et al. 2023a. Biomedgpt: A unified and generalist biomedical generative pre-trained transformer for vision, language, and multimodal tasks. *arXiv preprint arXiv:2305.17100*.

Mozhi Zhang, Mianqiu Huang, Rundong Shi, Linsen Guo, Chong Peng, Peng Yan, Yaqian Zhou, and Xipeng Qiu. 2024. Calibrating the confidence of large language models by eliciting fidelity.

Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, and Yang Feng. 2023b. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models.

Wenxuan Zhang, Sharifah Mahani Aljunied, Chang Gao, Yew Ken Chia, and Lidong Bing. 2023c. M3exam: A multilingual, multimodal, multilevel benchmark for examining large language models.

A Dataset Details

- **TriviaQA** (Joshi et al., 2017) is a reading comprehension dataset that features more than 650,000 question-answer-evidence triples. It consists of lots of question-answer pairs created by trivia aficionados, along with independently collected evidence documents—averaging six per question—that offer robust distant supervision for responding to the queries.
- **HaluEval** (Li et al., 2023a) is a benchmark designed to assess how well LLMs hallucinations—unverifiable or incorrect content in their outputs. It includes a collection of generated texts and human-annotated samples that help evaluate the models' performance in detecting such errors.
- **HalluQA** (Cheng et al., 2023) is a dataset consisting of 450 carefully crafted adversarial questions that cover various domains, incorporating elements of Chinese historical culture, customs, and social phenomena. It aims to evaluate LLMs on their propensity to produce two types of errors: imitative falsehoods and factual inaccuracies.
- **WebQA** (Li et al., 2016) is a large-scale, human-annotated real-world QA dataset, developed to address the scarcity of extensive real-world QA datasets for neural QA systems.

B Experiment Results

We show the full experiment results in Table 7, Table 8, and Figure 8.

C Details of Constructed Dataset

For the generated dataset, inspired by previous studies (Huang et al., 2023b; Yu et al., 2023), we employed attribute-guided prompting to instruct LLMs to generate relevant questions on specific topics, as illustrated in Table 9. We chose GPT-4 as our generation model because of its exceptional ability to follow instructions. The prompt template is shown in Figure 9. For the generated items, we

Dataset	Lang.	ChatGLM3		ChatGPT		GPT-4		Yi-34b		Qwen-turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
HalluEval	(en)	18.03%	18.03%	57.98%	57.84%	67.13%	66.99%	42.86%	41.75%	29.31%	29.59%	40.67%	40.67%
	(ch)	11.23%	17.34%	32.07%	52.38%	47.99%	65.05%	25.10%	39.81%	19.35%	26.51%	25.35%	37.33%
HalluQA	(en)	20.00%	25.41%	34.27%	30.90%	51.89%	53.51%	38.38%	47.03%	25.97%	37.57%	22.83%	18.48%
	(ch)	22.16%	22.70%	21.91%	25.28%	49.73%	51.89%	45.95%	45.95%	43.65%	39.78%	15.22%	20.65%
Chinese Domain	(en)	9.52%	21.21%	41.85%	42.73%	56.71%	57.58%	33.33%	57.58%	27.19%	48.25%	30.74%	24.24%
	(ch)	32.47%	25.54%	41.85%	42.29%	59.31%	58.44%	63.64%	57.14%	62.28%	62.28%	18.61%	22.51%
triviaQA	(en)	36.32%	36.38%	90.53%	90.31%	94.09%	93.93%	79.33%	78.90%	59.59%	59.47%	77.27%	77.05%
	(ch)	21.33%	32.22%	54.60%	83.33%	82.77%	92.29%	59.43%	76.27%	41.53%	53.55%	43.92%	66.32%
WebQA	(en)	28.51%	38.96%	59.08%	58.98%	67.70%	69.61%	57.07%	69.71%	49.48%	62.11%	50.00%	47.08%
	(ch)	48.69%	42.07%	57.35%	59.29%	72.52%	72.32%	76.93%	74.12%	71.12%	70.70%	37.02%	40.54%

Table 7: Six LLMs' performance on the setting without a low-resource detector.

Dataset	Lang.	ChatGLM3		ChatGPT		GPT-4		Yi-34b		Qwen-turbo		Llama3-Ch.	
		Origi.	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*	*Origi.*	*Impro.*
HalluEval	(en)	18.03%	18.31%	57.98%	57.70%	67.13%	66.57%	42.86%	42.72%	29.31%	29.17%	40.67%	40.25%
	(ch)	11.23%	18.03%	32.07%	56.02%	47.99%	66.16%	25.10%	42.58%	19.35%	28.61%	25.34%	39.97%
HalluQA	(en)	20.00%	25.95%	34.27%	32.02%	51.89%	53.51%	38.38%	46.49%	25.97%	38.67%	22.83%	21.74%
	(ch)	22.16%	23.78%	21.91%	23.60%	49.73%	51.35%	45.95%	44.86%	43.65%	41.44%	15.21%	15.76%
Chinese Domain	(en)	9.52%	32.03%	41.85%	41.85%	56.71%	59.31%	33.33%	61.04%	27.19%	58.33%	30.74%	19.48%
	(ch)	32.47%	30.74%	41.85%	41.41%	59.31%	58.44%	63.64%	60.17%	62.28%	58.33%	18.61%	18.61%
triviaQA	(en)	36.32%	35.78%	90.53%	89.09%	94.09%	93.54%	79.33%	78.73%	59.59%	58.80%	77.27%	76.12%
	(ch)	21.33%	35.94%	54.60%	89.15%	82.77%	93.22%	59.43%	78.18%	41.53%	58.41%	43.92%	74.92%
WebQA	(en)	28.51%	44.38%	59.08%	59.90%	67.70%	70.81%	57.07%	73.72%	49.48%	67.70%	50.00%	46.48%
	(ch)	48.69%	46.99%	57.35%	58.88%	72.52%	71.61%	76.93%	74.22%	71.12%	69.25%	37.02%	41.15%

Table 8: Six LLMs' performance on the setting without language selection.

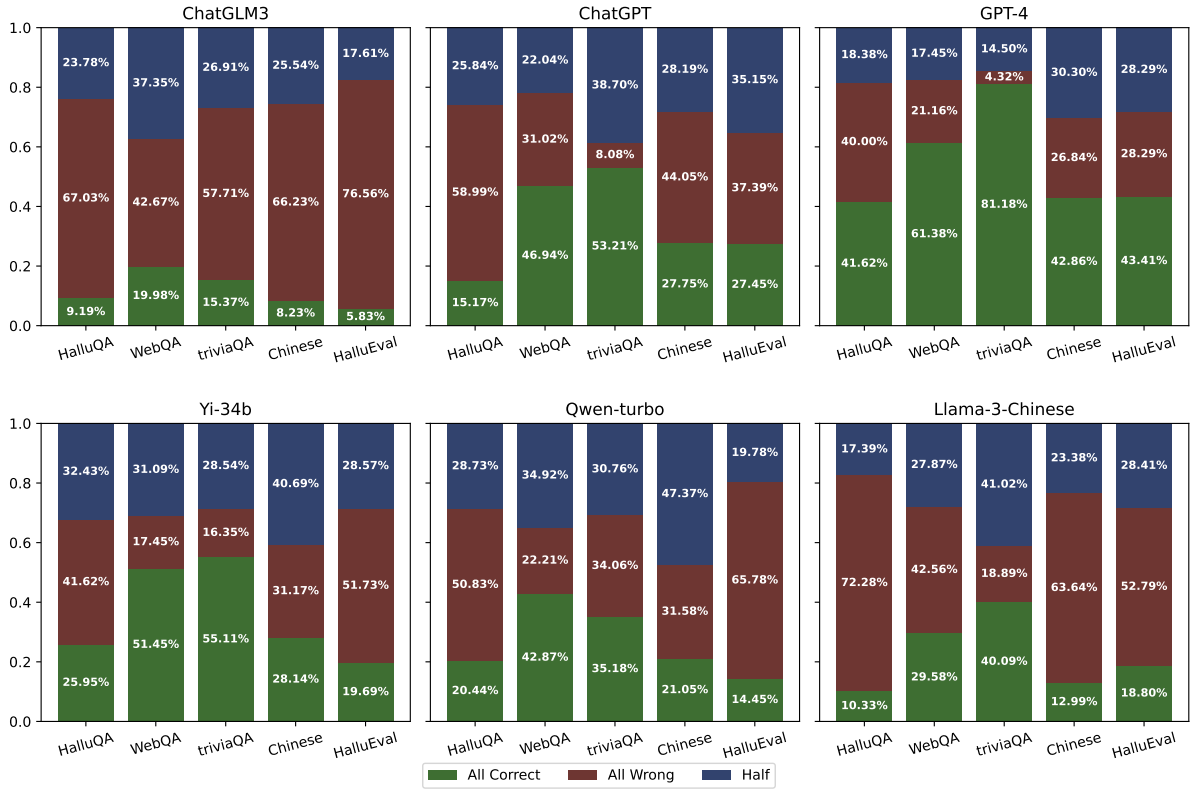

Figure 8: Performance percentage of LLMs across different datasets. 'All correct' indicates that the LLMs answered correctly in both the Chinese and English datasets. 'All wrong' signifies that the LLMs answered incorrectly in both datasets. 'Half' denotes that the LLMs answered correctly in only one of the datasets.

manually evaluate the correctness of its label to ensure the data quality.

D Collection of Chinese Domain Dataset

Our Chinese domain dataset consists of 227 items. This dataset encompasses knowledge and information specific to Chinese content, including aspects of geography, history, culture, and more. We sourced the content from a broad range of Chinese social media platforms and search engines. After initial curation, we conducted filtering to remove contents that cannot be accurately translated into English or may result in discrepancies in meaning upon translation, such as phrases from ancient Chinese.

E Answer Evaluation

We adopt the LLM-as-a-Judge for answer evaluation in all experiments to reduce the bias that comes from keyword matching. We use GPT-4 for our evaluation due to its exceptional capability and wide application in previous studies (Liu et al., 2023a; Gao et al., 2024). For the five QA datasets, we use the prompt template shown in Figure 10.

F Prompt Template

We show the prompt template used in our study in Figure 9, Figure 10, Figure 12, Figure 11, Figure 14, and Figure 13.

G Screenshots of Human Evaluation

We show the screenshots of human evaluation in Figure 15 and Figure 16.

Prompt Template

Next, I will provide you with a topic, and you will assist me in generating data based on this topic. I need you to generate three categories: questions with an English background, questions with a Chinese background, and questions with no specific language.
I will provide you with some examples:
Question: Piaget believes that communication has two functions, one is the egocentric function, and the other is?
Category: English knowledge
Question: With one byte, how many different codes can be generated at most?
Category: Knowledge with no specific language
Question: What are some famous dishes from Guangdong?
Category: Chinese knowledge
For each type of question, you need to generate ten, a total of thirty.
You need to return the data in JSON format, as follows:

```
{
"Question": "Category",
"Question": "Category",
"Question": "Category",
"Question": "Category",
...
}
```

Please generate the corresponding data in Chinese.
The topic I provide is: [TOPIC]

Figure 9: Prompt template for the generated dataset.

Prompt Template

As a helpful assistant, your task is now to help me assess the correctness of the provided answers. I will present a question along with its correct answer. Subsequently, I will also provide you with the answer you need to evaluate. If the answer to be evaluated correctly expresses the same meaning as the correct answer or contains the correct answer, then it is right. Ignore case errors. Although there are some errors in certain explanations within the answer, as long as the core answer is correct, the response is considered correct. Return me only one word: 'correct' or 'wrong'.
Here is the question and its correct answer:
Question: [QUESTION]
Answer: [ANSWER]
Here is the answer you should evaluate: [RES]

Figure 10: Prompt template for LLM-as-a-Judge.

Prompt Template

You are a very helpful assistant. I will provide you with a question and the answers in both Chinese and English. You need to integrate the Chinese and English answers to provide the final answer. During the integration process, you need to follow these rules:
1. You should primarily refer to the Chinese answer, appropriately integrating parts of the English answer.
2. If there is a factual conflict between the English and Chinese answers, you must refer to the Chinese answer.
3. The integrated answer should be of higher quality than the individual answers and better address the corresponding question.
4. The integrated answer must be all in English
Question: [[Q]]
Chinese answer: [[CH_RES]]
English answer: [[EN_RES]]

Figure 11: Prompt template for integration (For the situation when the selected language is English).

Prompt Template

你是一个非常有帮助的助手。我将给你提供一个问题，以及该问题的中英文的答案。你需要融合中英文答案，给出最终的答案。在融合答案的过程中，你需要遵循下面的规则：
1. 你需要着重参考英文的答案，适当融合部分中文的答案。
2. 如果英文的答案与中文的答案发生事实性冲突，你必须参考英文的答案。
3. 融合后的答案应该比融合前的答案具有更高的质量，更好地回答对应的问题。
4. 融合后的答案必须全都是中文。
问题：[[Q]]
中文答案：[[CH_RES]]
英文答案：[[EN_RES]]

Figure 12: Prompt template for integration (For the situation when the selected language is Chinese).

Prompt Template

As a helpful assistant, you need to categorize an English question, considering that the background of this question is not common in an English environment. Therefore, you need to choose the most suitable language for this question. You need to analyze the required language context for the question first, and then tell me at the end which language you think is most suitable to answer the question. The question is as follows:

Figure 13: Prompt template for language selection (For the query in English).

Prompt Template

作为乐于助人的助理，您需要将一个中文问题进行分类，考虑到该问题背景在中文环境中并不常见，因此您需要返回最适合该问题的语言。你需要首先对问题所需要的语言环境进行分析，然后在最后告诉我你返回的最适合回答该问题的语言。问题如下：

Figure 14: Prompt template for language selection (For the query in Chinese).

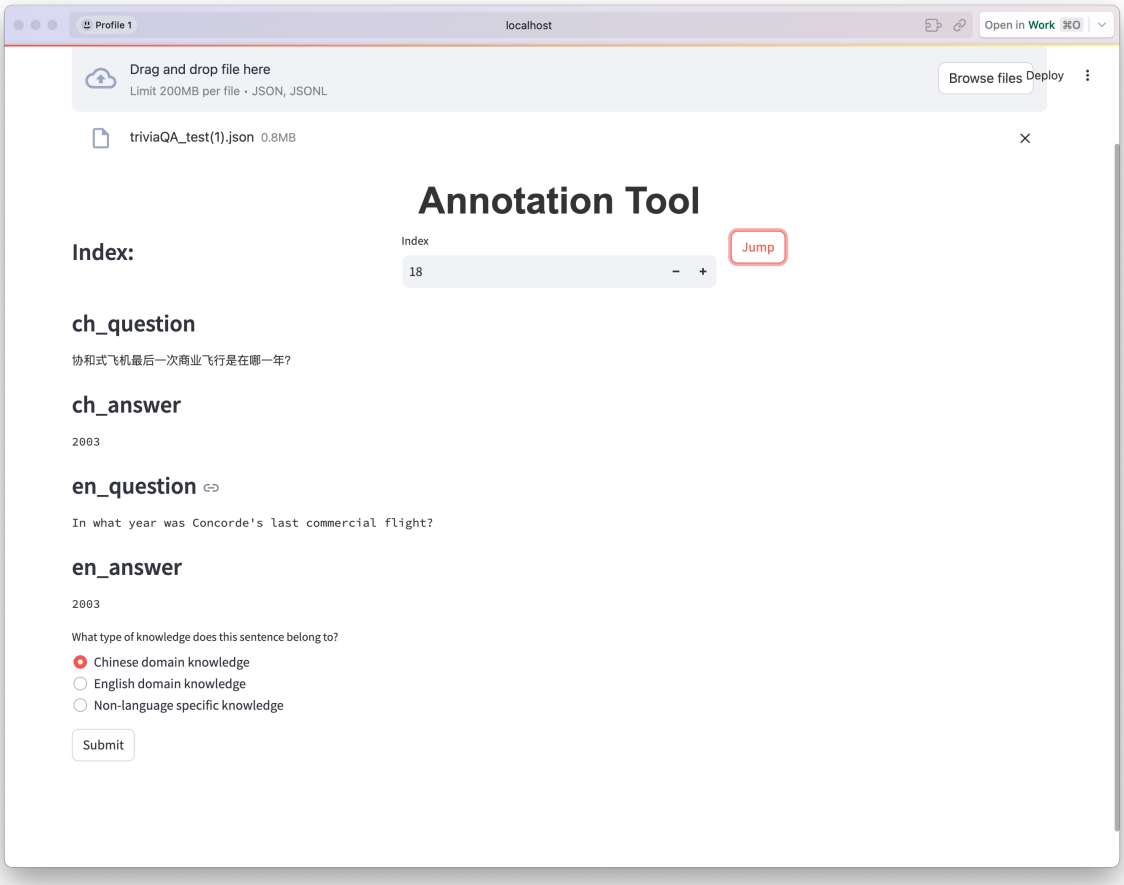

Figure 15: Screenshot of human annotation (1).

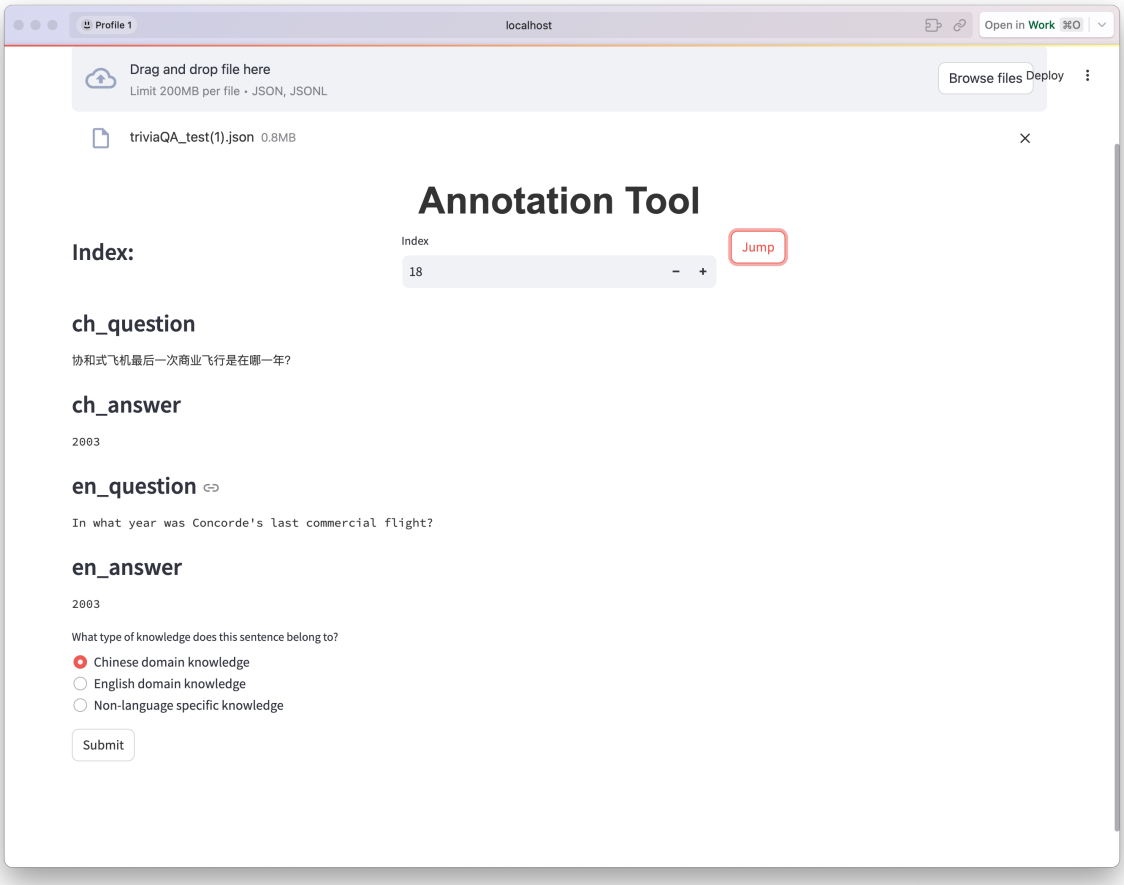

Figure 16: Screenshot of human annotation (2).

Table 9: Topics used for data generation.

Topic Word			
History	Literature	Science	Art
Social Sciences	Technology	Philosophy	Geography
Culture	Health	Artificial Intelligence	Machine Learning
Big Data	Blockchain	Internet of Things	Environmental Protection
Sustainable Development	Energy	Finance	Education
Human Genetics	Artificial Life	Space Exploration	Food Science
Sports	Psychology	Political Science	Economics
Sociology	Law		