Multilingual Retrieval-Augmented Generation for Knowledge-Intensive Question Answering Task

Leonardo Ranaldi **Barry Haddow** **Alexandra Birch**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
{first_name.last_name}@ed.ac.uk

Abstract

Retrieval-augmented generation (RAG) has become a cornerstone of contemporary NLP, enhancing large language models (LLMs) by allowing them to access richer factual contexts through in-context retrieval. While effective in monolingual settings, especially in English, its use in multilingual tasks remains unexplored.

This paper investigates the effectiveness of RAG across multiple languages by proposing novel approaches for multilingual open-domain question-answering. We evaluate the performance of various multilingual RAG strategies, including question-translation (**tRAG**), which translates questions into English before retrieval, and Multilingual RAG (**MultiRAG**), where retrieval occurs directly across multiple languages. Our findings reveal that **tRAG**, while useful, suffers from limited coverage. In contrast, **MultiRAG** improves efficiency by enabling multilingual retrieval but introduces inconsistencies due to cross-lingual variations in the retrieved content. To address these issues, we propose Crosslingual RAG (**CrossRAG**), a method that translates retrieved documents into a common language (e.g., English) before generating the response. Our experiments show that **CrossRAG** significantly enhances performance on knowledge-intensive tasks, benefiting both high-resource and low-resource languages.

1 Introduction

Retrieval-augmented generation (RAG) aims to improve the factuality and memory access of large language models (LLMs) by combining external knowledge during inference (Lewis et al., 2020b). RAG is designed to mitigate some of the well-known limitations of LLMs, including the tendency for hallucinations and the lack of specific domain knowledge in the training data (Siriwardhana et al., 2023; Kandpal et al., 2023).

Augmenting the questions by operating through relevant information retrieved from external corpora, such as Wikipedia effectively reduced inaccurate generation, thereby notably improving accuracies (Gao et al., 2024; Fan et al., 2024).

Nevertheless, previous efforts focused on English as the data language in their experiments, i.e., the language of the user queries and the retrieval corpora. Hence, limited attention is afforded to studying the type and role of non-English queries and retrieving multilingual documents to augment LLMs' capabilities. To address this gap, Zhang et al. (2022); Thakur et al. (2024a) proposed methodologies to evaluate multilingual retrieval.

On the other side of the coin, a series of studies initiated by Chirkova et al. (2024) have explored the impact of multilingual documents in the RAG pipelines, representing an early step towards understanding multilingual retrieval and generation.

In this paper, we systematically investigate the impact of RAG-based pipelines beyond English, aiming to identify potential challenges and propose strategies for improving the performance across a selection of languages. Taking previous work a step further, we evaluate the benefits of extending RAG methodologies in multilingual settings by analysing the effects of different types of retrieved documents on multilingual generative abilities in different languages. Complementing the previous works, we introduce and analyse the trade-off of different approaches that lead LLMs to harness multilingual knowledge.

This leads to the main research questions:
RQ1: How does multilingual retrieval affect RAG accuracy and consistency?
RQ2: What are the benefits and limitations of incorporating multilingual knowledge in RAG?
RQ3: Which methods could improve multilingual RAG performance?

To answer these answers, we produce a comprehensive evaluation by introducing strategies

Findings of the Association for Computational Linguistics: EACL 2026, pages 697–716
March 24-29, 2026

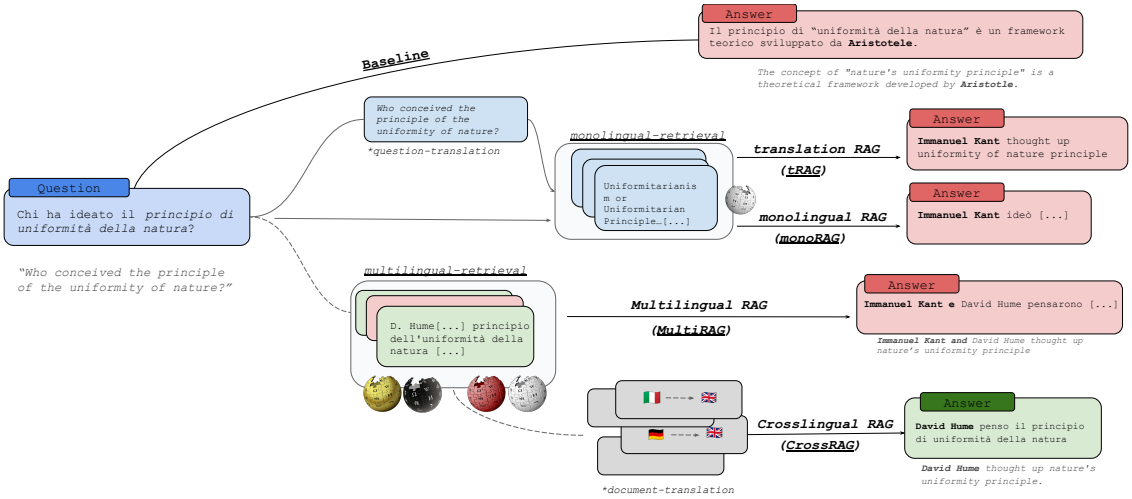


Figure 1: Retrieval-Augmented Generation (RAG) pipelines studied in our works. We explore the performances of different prompting pipelines to handle multilingual queries (§2).

for handling multilingual queries as well as retrieval stages, as shown in Figure 1. We use three knowledge-intensive question-answering tasks properly constructed for multilingual evaluation as they best represent multilingual open-ended question-answering tasks. We then operated different LLMs, chosen for proficiency in RAG tasks and multilingual performances, to investigate their capabilities in leveraging multilingual retrieved knowledge. The main contributions of our paper are:

- We explore RAG beyond English by showing the benefits derived from extending the range of retrieval to multilingual contexts. Firstly, we show that naïve approaches such as query translation (**tRAG**) generate incorrect translations, leading to wrong retrieval and misleading generations, for instance, Appendix P. Instead, Multilingual RAG (**MultiRAG**), based on multilingual retrieval and language-specific queries, outperforms monolingual RAG (**monoRAG**) based on monolingual retrieval sources in the query language.
- We then study the dynamics between the languages that emerge in **MultiRAG**. We outline the advantages of document retrieval over heterogeneous knowledge bases, and at the same time, we display the problems that some models have when they have to operate with retrieved knowledge from documents in different languages. Specifically, we show that the LLMs used in our work are proficient at understanding multilingual questions but fail in extracting information, especially in low-resource languages.
- In order to address these problems with **MultiRAG**, we introduce a document-level translation pipeline (**CrossRAG**) that allows the LLMs to handle knowledge-intensive tasks by operating with retrieved documents in a single language (i.e. English) but still providing multilingual responses.

2 Methods

Retrieval-augmented Generations (RAG) methods improve performance of LLMs in knowledge-intensive tasks by combining questions with retrieved knowledge in context (§2.1).

Although the usefulness of RAG has been demonstrated, evaluations and further studies are primarily conducted in English, leaving other languages unexplored. Hence, we propose a systematic study of the portability of RAG pipelines to languages other than English (§2.2), analysing different approaches to improve the effective value of expanding retrieval beyond English (§2.3).

2.1 RAG Pipelines

In traditional RAG, knowledge is acquired from domains $\mathcal{D}$ (e.g., Wikipedia or internal databases) and used during inference to promote accurate generation. The pipeline is structured into phases:

Retrieval In this phase, the relevant top-k documents $docs = \{d_1, \ldots, d_k\}$ are retrieved, based on the query $\mathcal{Q}$ from $\mathcal{D}$, using a retrieval system $\mathcal{R}$.

RAG Prompt Template

Please answer the question by following the provided instructions.
#Instructions:
Answer the question as clearly as possible using the provided ***reference evidence*** *and follow the format 'Answer:'*
#Reference Evidence:
$docs = \text{top-k}\{h_{\mathcal{Q},\mathcal{C}}\}, \mathcal{C} \in \mathcal{D}$
#Question: $\mathcal{Q}$

Table 1: RAG instructions (prompt) for the model to elicit they to consider the reference evidence (*docs*) for generating an answer to a given question ($\mathcal{Q}$).

During retrieval with $\mathcal{R}$, the question $\mathcal{Q}$ and documents in $\mathcal{D}$ are encoded, forming $h_{\mathcal{Q}} = \mathcal{R}(\mathcal{Q}) \in \mathbb{R}^n$ for query and $h_{\mathcal{D}} = \mathcal{R}(\mathcal{D}) \in \mathbb{R}^n$ for documents. Then, the similarity $\langle h_{\mathcal{Q}}, h_{\mathcal{D}} \rangle$ is used to select a collection $\mathcal{C}$ from $\mathcal{D}$, consisting of documents that best match the query. Usually, to improve retrieval quality, the top most relevant documents are filtered and ordered using a re-ranked model obtaining $docs$. Then, they are encoded together with the query in $h_{\mathcal{Q},\mathcal{C}} = \mathcal{R}(\mathcal{Q}, \mathcal{C}) \in \mathbb{R}$. This allows for $h_{\mathcal{Q},\mathcal{C}}$ representations, which capture similarities between the query and the documents by improving quality and considering similarity, specific contexts, and semantic relevance to $docs = \text{top-k}\{h_{\mathcal{Q},\mathcal{C}}\}$.

The retriever generally uses a ranker based on architectures trained on specific information retrieval datasets and a customised reranker. Since our work focuses on using such systems in a multilingual setting, we operate via retriever and re-ranker systems provided by Cohere detailed in §3.2.

Inference The second phase consists of augmenting the LLMs' capabilities in answering a given query $\mathcal{Q}$ using knowledge delivered from reference evidence, i.e. the retrieved relevant documents. The LLM generates the answer $\mathcal{A}$ from $\mathbb{LLM}(\mathcal{Q}, docs)$. Here, using a well-defined prompt *template* to get the LLM to consider retrieved documents a source of knowledge is recommended. Following the earlier RAG heuristics, we propose a standard template that instructs the model ("**#Instructions**") to consider "**#Reference Evidence**" in retrieved documents for delivering the final answer $\mathcal{A}$. An example prompt is reported in Table 1.

2.2 RAG beyond English

Monolingual RAG (`monoRAG`) In general RAG pipelines (§2.1), it is assumed that the query $\mathcal{Q}$ and $\mathcal{A}$ are in the same language, which we refer to as L_{SL}. Consequently, the $docs$ retrieved from $\mathcal{D}_{SL}$ are in L_{SL}. Hence, in this setting, we instruct the $\mathbb{LLM}$ using the template in Table 1, $docs = \text{top-k}\{h_{\mathcal{Q}_{SL},\mathcal{C}}\}$ *where* $\mathcal{C} \in \mathcal{D}_{SL}$ and $\mathcal{Q}_{SL}$ is in L_{SL}. For the rest of the paper, we refer to this setting as monolingual RAG (**`monoRAG`**).

Translation RAG (`tRAG`) Since the knowledge sources from which retrieval is made are generally richer in English (Sharma et al., 2024), a practical way to solve RAG beyond English is to translate the $\mathcal{Q}_{SL}$ to English $\tilde{\mathcal{Q}}_{En}$ using a translation system and perform the retrieval from $\mathcal{D}_{En}$. Then, we instruct the $\mathbb{LLM}$ using the same setting of **`monoRAG`**, but in contrast, the retrieved documents are $docs = \text{top-k}\{h_{\tilde{\mathcal{Q}}_{En},\mathcal{C}}\}$ where $\mathcal{C} \in \mathcal{D}_{En}$.

Although these strategies can solve the language barrier by allowing non-English queries to operate, the scope of retrieval is limited to only one domain, namely $\mathcal{D}_{En}$ for **`tRAG`** and $\mathcal{D}_{SL}$ for **`monoRAG`**. In addition to the limited scope, the translation also affects retrieval (for instance, see the example in Appendix P).

Multilingual RAG (`MultiRAG`) Hence, we extend the scope of the retrieval to many languages. We use a retriever whose database consists of $\bigcup_{\mathcal{D}_i \in L}$, i.e. the union across all resources in all available languages. As in the **`monoRAG`**, we instruct the $\mathbb{LLM}$ to consider the retrieved documents using the template in Table 1. In contrast to the previous approaches we use the $docs = \text{top-k}\{h_{\mathcal{Q}_{SL},\mathcal{C}}\}$, where $\mathcal{C} \in \bigcup_{\mathcal{D}_i \in L}$ and $\mathcal{Q}_{SL}$.

2.3 Cross-lingual RAG

Multilingual retrieval and prompting strategies broaden the scope of retrieval. As a result, retrieved documents can be in any language in $\mathcal{D}$.

Although this is a plus for retrieval, it can degrade the LLM's responses, for example, generating answers in the wrong language (see example in Appendix R) because it must combine in-context documents in different languages.

To solve this issue, we propose Cross-lingual RAG (**`CrossRAG`**), in which the documents are retrieved as in **`MultiRAG`** but are then translated by an external tool $\mathbb{T}$ and delivered at inference time in English. This approach improves the accu-

racy of the response without requiring a substantial additional computational effort.

3 Experiments

We select three multilingual open-domain question-answering tasks (§3.1) to compare our approaches. We perform the retrieval and inference phases described in §3.2 and perform the evaluations as presented in §3.3.

3.1 Tasks & Datasets

We use the following question-answering (QA) tasks: *(i)* MLQA (Lewis et al., 2020a), *(ii)* MKQA (Longpre et al., 2021) and *(iii)* XOR-TyDi QA (Asai et al., 2021) as they best represent multilingual open-ended question-answering tasks. These datasets are extensions of resources that originated in English. MLQA and MKQA are manually and machine-translated, whereas XOR-TyDi QA is translated by professional annotators. We provide details about the languages covered and the number of questions in Appendices F and I.

3.2 Experimental Setup

To explore RAG pipelines beyond English, we apply the methods introduced in §2 based on retrieval and inference phases.

Retrieval We use Cohere as the retrieval system $\mathcal{R}$ and Wikimedia_dump as the database $\mathcal{D}$ for all experiments[1]. Specifically, in the version[2] provided by Cohere, individual articles are embedded with multilingual embedding model *Cohere_Embed_V3* (we report the dump composition in Table 7). Following the approaches proposed by Asai et al. (2023); Chirkova et al. (2024); Ranaldi et al. (2025d), we retrieve the most relevant passages and use top-5 as in-context knowledge during inference (details in Appendix D). As described in §2.2 we either use *(i) monolingual retrieval* (**monoRAG** and **tRAG**) which consists of retrieval on $\mathcal{D}_{SL}$ with documents only in a single specific language, or *(ii) multilingual retrieval* (**MultiRAG** and **CrossRAG**) which consists of retrieval on $\bigcup_{\mathcal{D}_i \in L}$ that is the union of multiple $\mathcal{D}_i$ in L used in the evaluated task.

Prompting We instruct the LLMs using the prompts introduced in §2. We then include explicit instructions that elicit the model to consider the input query (**#Question:**), retrieved documents (**#Reference Evidence:**) and deliver the final answer in an **"evaluated language"** that, by the construction of our experiments, corresponds to the query language.

Translation As a translation system, in the main discussion, we use `Google Translate`[3] to translate for both **tRAG** and **CrossRAG**. Furthermore, we investigate the effect of using LLMs as translation systems operating via GPT-4o and other approaches detailed discussed in §4.

Models & Inference Settings To get a comprehensive evaluation of existing RAG pipelines, we use three different LLMs: GPT-4o (OpenAI, 2023), Llama-3-8b-instruct (Touvron et al., 2023) and Command-R-35b[4] (Cohere Inc., 2024). Detailed settings and model versions are in Appendix A. We use greedy decoding in all experiments to ensure a more deterministic generation process. We set most deterministic temperatures to 0 and the maximum generation length to 2048.

3.3 Evaluation

We use flexible exact-match accuracy following Schick et al. (2023); Mallen et al. (2023), which is based on whether or not ground-truth answers are included in the generated answers provided by the models instead of a strict exact match. Furthermore, for a complete comparison, we follow Chirkova et al. (2024) to conduct multilingual evaluations using the SQUAD evaluation script and 3-gram character level.

4 Results & Discusssions

The empirical results across different languages on MKQA, MLQA and XOR TyDi QA are reported in Figure 2. Overall, the experiments confirm that extending the retrieval scope to multilingual contexts (**MultiRAG**) improves the RAG-based pipelines, outperforming language-specific monolingual RAG (**monoRAG**) and naïve approaches that address the language barrier using query translation (**tRAG**). Indeed, although monolingual retrieval (i.e., **monoRAG**) achieves benefits compared to the baseline, the retrieved documents may be limited and consequently could not contain

[1]This pipeline makes it easy to search Wikipedia for information and to restrict it to specific languages.

[2]Cohere/wikipedia-2023-11-embed-multilingual-v3

[3]used via Google Translate API python package

[4]To simplify discussion for the rest of the paper, we will refer to these models using Llama-3-8b and Command-R.

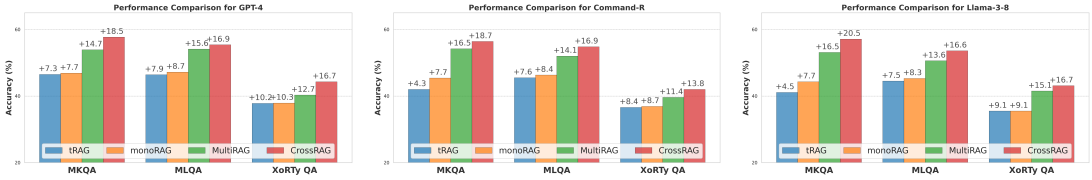


Figure 2: Performance comparison of models using RAG approaches described in §2 across benchmarks and settings detailed in §3, separated by average (Avg), high-resource (HR) and low-resource (LR) languages averages. The values above the bars are the differences with the baselines (no RAG scores).

the necessary information to answer a language-specific query. Conversely, retrieval from multilingual heterogeneous sources has a broader range of results. However, multilingual knowledge could lead models to wrong generations (see the example in Appendix R). Therefore, we proposed **CrossRAG** to harness **MultiRAG** retrieval by operating with documents in the same language, i.e. English.

In the following sections, we analyse the benefits a multilingual retrieval brings when adopted in a RAG strategy (§4.1), then we examine the effects across different languages §4.2 and propose two strategies to improve the practical usage of the retrieved knowledge in multilingual settings §4.3. Finally, we conduct additional studies by investigating the role of the retriever and its impact on the final performances (§5.2), the generated languages (§5.1) and the robustness on challenging perturbations (§5.3).

4.1 The impact of RAG beyond English

Figure 2 shows the results obtained from different LLMs when prompted with RAG-based strategies in monolingual and multilingual settings as introduced in §2. An overall improvement over baseline models without RAG can be observed using monolingual RAG, i.e., **monoRAG** (+8.9% improvement for GPT-4o, +7.9% improvement for Llama-3-8b and +8.2% improvement for Command-R). Moreover, the results show that the impact of extending retrieval to multilingual settings and using retrieved passages in RAG-based approaches (**MultiRAG** strategy) brings clear benefits. Indeed, performance consistently increases compared to the **monoRAG** (+5.4% for GPT-4o, +7.1% for Llama-3-8b and +5.7% for Command-R). This indicates that multilingual retrieval provides access to broader information that could be unavailable in monolingual resources; for instance, in the example reported in Appendix Q, the information about "England Queens" are not available in Chinese Wikipedia. However, since the scope of retrieval is wider and the retrieval languages are multiple, the languages of the retrieved documents may impact the performance differently, as discussed in §4.2.

4.2 Knowledge Diversity

Multilingual RAG (**MultiRAG**) shows average improvements compared to the baselines, **monoRAG** and **tRAG**, where the retrieved documents are in a single language as discussed in §4). In **MultiRAG**, the knowledge retrieved are in different languages (as reported in Figure 3, these are average documents retrieved per language). The differences in percentages are due to the composition of the Wikimedia_dumps (reported in Table 7) undersized for some languages.

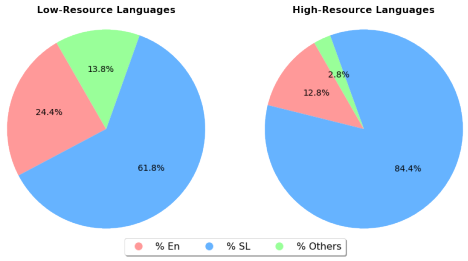


Figure 3: Average percentage languages of retrieved documents (details in Appendix 13).

Consequently, the effect of **MultiRAG** is different even between languages. Figure 4 shows that the effect of **MultiRAG** on low- (LR) languages is more marked than high-resource (HR) languages[5]. Indeed, comparing **MultiRAG** with **monoRAG** in the case of HR, we observe average increases of +3.6% for GPT-4o, +4.1% for Llama-3-8b and +4.4% for Command-R. In contrast, for LR, there is an average increase of 6.6% for GPT-4o, 8.4% for Llama-3-8b and 7.7% for Command-R).

To gain a comprehensive view of the role of multilingual retrieval conducted in the **MultiRAG**

[5]high- and low-resources explained in Appendix B

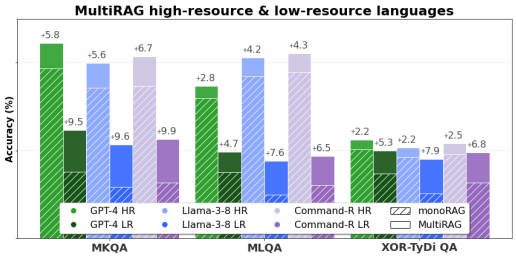


Figure 4: Accuracies **monoRAG** and **MultiRAG** in low- (LR) and high-resource (HR) languages. *(differences are above the bars).

setting, we performed further experiments by restricting the retrieval to a set formed by specific language (SL) and English, which we define as *(En+SL)*. Figure 8 (Appendix M) shows the differences in terms of performance when the scope of the retrieval of **MultiRAG** is broader, i.e. all languages available in the dump used in the experiment (Wikipedia versions as detailed in §3) and *(En+SL)*. On average, extending the scope of retrieval beyond the subset represented of *En+SL* has benefits except GPT-4o in the MKQA task and Llama-3-8b in the case of MLQA.

Hence, although **MultiRAG** consistently achieves higher performance than **monoRAG**, there are some cases where the heterogeneity of languages is not beneficial. For instance, the case in Appendix R where passages in English is not taken into account by the model. Therefore, to analyse whether the component affecting performances is the ability to leverage the different languages in different retrieved documents, we propose an intervention strategy by introducing a translation phase of the retrieved multilingual knowledge and discuss the results in §4.3.

Model		Δ MKQA	Δ MLQA	Δ XoR Ty-QA	Average
GPT-4o	Avg	+3.8	+1.3	+5.5	+3.5
	HR	**+4.2**	+0.7	+2.7	+2.5
	LR	+1.8	**+2.1**	**+7.1**	**+3.7**
Command-R	Avg	+2.2	+2.8	+1.6	+2.2
	HR	+2.7	+3.7	+3.9	+3.4
	LR	**+5.2**	**+4.3**	**+5.5**	**+5.0**
Llama-3-8b	Avg	**+4.0**	+3.2	+1.6	+2.9
	HR	+1.8	+2.4	+3.9	+2.7
	LR	+3.7	**+4.2**	**+4.6**	**+4.1**

Table 2: Differences (Δ) between **CrossRAG** and **MultiRAG**.*In **bold**, the highest differences for model.

4.3 When translating matters

The red bars in Figure 2 show that the average results obtained by **CrossRAG** are consistently better than those of other approaches. In general, translating the retrieved information into English benefits the final performance[6]. In Table 2, we report the performance improvements over **MultiRAG** differentiated for LR and HR. Here, we observe that in HR, there are improvements of around +2.5 for GPT-4o and Llama-3-8b and +3.4 for Command-R when compared to **MultiRAG**. In contrast, we note larger benefits for LR (respectively +3.7 for GPT-4o, +5 for Command-R and +4.1 for Llama-3-8b on average). These results highlight the limitations that the LLMs examined have when operating via **MultiRAG** concerning documents in multiple languages (see the case discussed in §4.2 in Appendix R).

However, since the translation component matters, we proposed the same experimental setting using *(i)* GPT-4o as the translation tool, *(ii)* instruction-tuning at the translation level.

Model		tRAG	tRAG	CrossRAG	CrossRAG
GPT-4o	MKQA	46.5	**48.3**	60.4	**62.0**
	MLQA	46.4	47.9	55.4	**58.8**
	XoR TDQA	37.7	38.2	45.8	**49.3**
Command-R	MKQA	39.9	40.3	56.4	57.8
	MLQA	45.5	46.0	54.8	56.2
	XoR TDQA	36.6	37.3	42.0	**44.5**
Llama-3-8b	MKQA	41.4	42.0	57.2	58.5
	MLQA	44.5	44.6	53.6	55.4
	XoR TDQA	37.0	37.7	44.5	**47.3**

Table 3: Average performances using two different translation systems. *In **bold**, the differences that exceed at least 2 points. **XoR TiDy-QA (XoR TDQA)

GPT-4o as translator Here, we propose different settings to observe the effect of various systems on the performance of our **CrossRAG**. Hence, we used GPT-4o (`GPT-4o` as in §3.2) as the translation tool. Then, using the prompt in Appendix K, we translated both retrieved documents and questions (in two distinct experimental phases) and reproduced the experimental setting proposed earlier. Table 3 compares the results using two different

[6]In Appendix U, we experimented with translation into languages other than English.

systems. In the case of **tRAG**, there are no conspicuous improvements (highest difference +1.9 in GPT-4 MKQA). Concerning **CrossRAG**, it can be observed that significant differences emerge between the final results achieved by using Google Translate and GPT-4o. This further demonstrates *(i)* the importance of multilingual retrieval (greater range of retrieval) and *(ii)* the usability of retrieved knowledge by LLM is better when it is in English. Indeed, multilingual knowledge retrieved and then processed in English impacts the final generations, whereas the same knowledge (the same docs in a foreign language) does not have the same impact.

Method		MKQA	MLQA
MultiRAG	**Avg**	53.1	50.6
	LR	44.0	38.7
CrossRAG	**Avg**	57.2	53.8
	LR	46.7	41.9
TF	**Avg**	**58.9**	54.7
	LR	45.2	**43.6**
CrossRAG (GPT-4o)	**Avg**	58.5	**55.4**
	LR	47.3	42.8

Table 4: Evaluation using Translation-following (TF), **MultiRAG** and **CrossRAG** (with Google Translate and GPT-4o as translation tools) on Llama-3-8b.

Translation-following Since the language of the retrieved documents plays a crucial role in the model's performance, we propose a multilingual augmentation strategy conceived to enhance their capability to operate on multilingual documents. Hence, we employ the Translation-following (TF) approach as proposed in our related work (Ranaldi et al., 2024; Ranaldi and Freitas, 2024) and detailed in Appendix G. Table 4 shows that Llama-3-8b enhanced through the TF achieves consistent benefits. In particular, 11.6% on MKQA and 7.5% on MLQA on average values when compared with **MultiRAG**. While 4.9% on MKQA and 1.8% on MLQA on average values when compared with **CrossRAG**. Finally, when compared with the **CrossRAG** version with GPT-4o as a translation tool, it achieves comparable performance (differences around <1). However, although performance increases are evident in some cases, there is the cost of additional tuning that should be considered.

5 Ablation Analysis

The results discussed in §4 show the benefits of *(i)* extending retrieval beyond English contexts and *(ii)* the operability of in-context approaches and translation tools to align the language of different retrieved information. This section analyses the qualitative impact of the proposed techniques on generations (§5.1) and the effect of using other retrievals (§5.2). Finally, in §5.3, we study the robustness of LLM to the combination of information in different languages and the number of documents retrieved.

5.1 Language Generated

One of the requirements for the correct answer is that the language must be the same as the query (the labels are also in a specific language). As an evaluation metric, in addition to the accuracy discussed in §5, we evaluate the percentage of answers generated in the correct language. To do this, we use the OpenLID framework (Burchell et al., 2023). Figure 5 shows that **CrossRAG** achieves consistently higher rates than **MultiRAG**. Moreover, **monoRAG** gets comparable performances to the baseline (when we do not use RAG approaches), but the accuracy is significantly lower. These results demonstrate that the models analysed generally follow instructions (given prompt); however, when operating with multilingual knowledge (i.e., **MultiRAG**), they fail to both follow instructions and deliver the correct response, especially in low-resource languages.

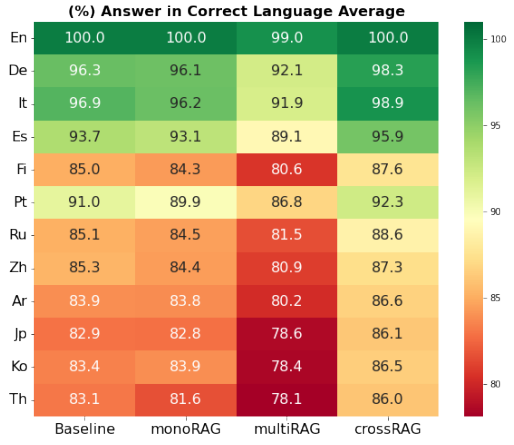


Figure 5: Average generated languages for MKQA. Appendix N reported detailed results.

5.2 Retrieval Settings

In our experimental setting (§3), we use Cohere as a retrieval tool. To observe the impact of the retrieval methodologies on the performances, we conducted a parallel experiment using BGE-m3 as in (Chirkova et al., 2024) (detailed in Appendix E). Figure 6 shows the average performances obtained by Llama-3-8b on the MKQA subset using the two different retrieval strategies. There are no conspicuous differences on average. This indicates that although the retrieval techniques differ, they provide equivalent retrieval methodologies. We use Cohere because it allows for an already-indexed version of the Wikipedia dump, as discussed in Appendix D.

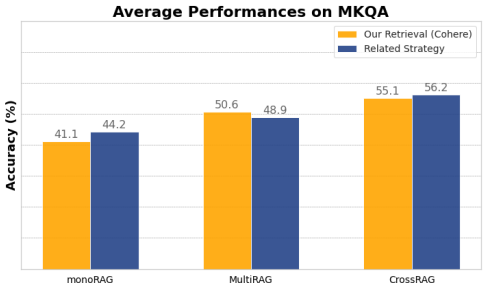


Figure 6: Retrieval strategy differences (overlap in Appendix T).

5.3 Robustness Analysis

Figure 7 shows a robustness analysis of the proposed approaches. We analysed the impact of the order of the retrieved documents by selecting the knowledge provided at the inference phase and conducting an extensive retrieval and a re-ranking (details in §3.2). To observe the impact of the order of the provided knowledge (documents), we *(i)* randomly shuffled documents (Random Shuffle), *(ii)* English documents in first positions (En doc/s first), *(iii)* English documents at the last positions (En doc/s last). From the results in Figure 7, it emerges that **`CrossRAG`** is robust to the varying document order. Instead, **`MultiRAG`** is more sensitive to retrieval order; this phenomenon emerges in Llama-3-8b and less markedly in Command-R and GPT-4o. This further indicates that the language sensitivity of documents is a drawback to the final performance, and operating a translation process or system such as **`CrossRAG`** improves performance by making it more robust to scenarios where retrieved documents may not be delivered in the most optimal order.

6 Related Work

Previous research investigated the advantages of augmenting LLMs through retrieved knowledge, a technique known as Retrieval-augmented Generative (RAG) (Lewis et al., 2020b). Many efforts have concentrated on exploring techniques to improve RAG by operating in-context (Menick et al., 2022), tuning (Gao et al., 2023), or intervening on retrievers (Sawarkar et al., 2024). Although these results represent a considerable step forward, little attention has been paid beyond English. We work on multilingual tasks involving multilingual queries and documents in the evaluation. While the tremendous effort of Zhang et al. (2022); Thakur et al. (2024b) is focused on the study of retrieval from multilingual sources and proposed benchmark-related, we study the role that retrieved documents have on the inference phase of LLMs. Enriching the foundation work proposed by Chirkova et al. (2024), we study the impact that different architecture components have on final performance. We analyse the effect of translation at different levels (before and after retrieval) using various tools across high- and low-resource languages. Analysing criticisms and strengths of Multilingual RAG, we study the roles of different solutions, showing when they lead the LLMs to leverage multilingual knowledge by obtaining consistent benefits.

7 Recommendations & Future Work

The experiments conducted aim to measure the impact of RAG-based approaches in multilingual settings. The recommendations that emerged from our analysis are: *(i)* Expanding retrieval beyond English-only resources benefits the performance of multilingual knowledge-intensive tasks when they are addressed with RAG-based pipelines. *(ii)* The components for solving multilingual tasks in RAG scenarios have a different impact depending on positioning (e.g. translation systems in **`tRAG`** and **`CrossRAG`**). *(iii)* As a consequence of *(ii)*, conducting multilingual retrieval and translating the retrieved knowledge consistently improves performance, as putting the information all on the same plane (language) leads LLMs to better understand and deliver the correct answer.

However, one of the limitations we will investigate in future developments is that information across different languages may not be aligned in LLMs or in known knowledge dumps (Ranaldi,

2025; Ranaldi et al., 2025a). In this regard, in our follow-up, we investigated different discrepancies and proposed techniques for manipulating and transferring knowledge in English (Ranaldi et al., 2025d) and in a multilingual scenario (Ranaldi et al., 2025c) as well as reasoning capabilities (Ranaldi and Pucci, 2025). Although this work has demonstrated the effectiveness of the proposed approach, limitations remain: the models are highly susceptible to changes in prompts in multilingual scenarios, especially when non-parametric knowledge is present (Pucci and Ranaldi, 2025). This latest phenomenon opens the way to further applications, once again highlighting the strategic importance of these architectures and studies.

8 Conclusion

RAG has shown great potential in boosting the performance of LLMs on knowledge-intensive tasks. Yet, scenarios beyond English represent a significant limitation. We proposed strategies to mitigate these restrictions by introducing retrieval expansion techniques and interventions on retrieved documents. We then analysed the performance of different LLMs in multilingual tasks. The results demonstrate that multilingual retrieval outperforms monolingual and greedy query translation approaches. This research underscores the need to understand RAG pipelines beyond English to ensure reliable multilingual access and improve model performance across diverse languages.

Acknowledgements

This work is funded by EU Horizon Europe (HE) Research and Innovation programme grant No 101070631, and UK Research and Innovation under the UK HE funding grant No 10039436.

Limitations

Due to the limitations imposed by the evaluation benchmarks and the cost of the closed-source models, we conducted tests on three tasks in different languages, which only scratches the surface of the world's vast array of languages. In the future, it will be appropriate to study the generality of our approach compared to other closed-source large language models. Additionally, it would be interesting to investigate the impact of using additional knowledge bases such as DBpedia.

Ethics Statemet

In our work, ethical topics were not addressed. The data comes from open-source benchmarks, and statistics on language differences in commonly used pre-training data were obtained from official sources without touching on gender, sex, or race differences.

References

Akari Asai, Jungo Kasai, Jonathan Clark, Kenton Lee, Eunsol Choi, and Hannaneh Hajishirzi. 2021. XOR QA: Cross-lingual open-retrieval question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 547–564, Online. Association for Computational Linguistics.

Akari Asai, Sewon Min, Zexuan Zhong, and Danqi Chen. 2023. Retrieval-based language models and applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts)*, pages 41–46, Toronto, Canada. Association for Computational Linguistics.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 865–879, Toronto, Canada. Association for Computational Linguistics.

Nadezhda Chirkova, David Rau, Hervé Déjean, Thibault Formal, Stéphane Clinchant, and Vassilina Nikoulina. 2024. Retrieval-augmented generation in multilingual settings.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Cohere Inc. 2024. Command-r documentation. `https://docs.cohere.com/v2/docs/command-r`. Accessed: 2024-12-08.

Common Crawl. 2021. Common crawl 2021. Web. Accessed: 2023-12-12.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent

Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020a. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. 2022. Teaching language models to support answers with verified quotes.

OpenAI. 2023. Gpt-4 technical report.

Giulia Pucci and Leonardo Ranaldi. 2025. Advancing oversight reasoning across languages for audit sycophantic behaviour via X-agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 12949–12965, Suzhou, China. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Federico Ranaldi, Andrea Zugarini, Leonardo Ranaldi, and Fabio Massimo Zanzotto. 2025a. Protoknowledge shapes behaviour of llms in downstream tasks: Memorization and generalization with knowledge graphs.

Leonardo Ranaldi. 2025. Survey on the role of mechanistic interpretability in generative ai. *Big Data and Cognitive Computing*, 9(8).

Leonardo Ranaldi and Andre Freitas. 2024. Aligning large and small language models via chain-of-thought reasoning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1812–1827, St. Julian's, Malta. Association for Computational Linguistics.

Leonardo Ranaldi, Barry Haddow, and Alexandra Birch. 2025b. When natural language is not enough: The limits of in-context learning demonstrations in multilingual reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 7369–7396, Albuquerque, New Mexico. Association for Computational Linguistics.

Leonardo Ranaldi and Giulia Pucci. 2025. Multilingual reasoning via self-training. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11566–11582, Albuquerque, New Mexico. Association for Computational Linguistics.

Leonardo Ranaldi, Giulia Pucci, Barry Haddow, and Alexandra Birch. 2024. Empowering multi-step reasoning across languages via program-aided language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12171–12187, Miami, Florida, USA. Association for Computational Linguistics.

Leonardo Ranaldi, Federico Ranaldi, Fabio Massimo Zanzotto, Barry Haddow, and Alexandra Birch. 2025c. Improving multilingual retrieval-augmented language models through dialectic reasoning argumentations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9064–9085, Suzhou, China. Association for Computational Linguistics.

Leonardo Ranaldi, Marco Valentino, and Andre Freitas. 2025d. Eliciting critical reasoning in retrieval-augmented generation via contrastive explanations. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11168–11183, Albuquerque, New Mexico. Association for Computational Linguistics.

Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. 2024. Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Nikhil Sharma, Kenton Murray, and Ziang Xiao. 2024. Faux polyglot: A study on information disparity in multilingual large language models.

Shamane Siriwardhana, Rivindu Weerasekera, Elliott Wen, Tharindu Kaluarachchi, Rajib Rana, and Suranga Nanayakkara. 2023. Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17.

Nandan Thakur, Luiz Bonifacio, Xinyu Zhang, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, and Jimmy Lin. 2024a. Nomiracl: Knowing when you don't know for robust multilingual retrieval-augmented generation.

Nandan Thakur, Suleman Kazi, Ge Luo, Jimmy Lin, and Amin Ahmad. 2024b. Mirage-bench: Automatic multilingual benchmark arena for retrieval-augmented generation systems.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2022. Making a miracl: Multilingual information retrieval across a continuum of languages.

A Models Vesions

For all our experiments, we use the versions of the models and datasets published on HuggingFace (via API for GPTs). All artefacts are released under the Apache-2.0 licence and MIT licence (reported in the official repositories).

Model	Version
GPT-4o	OpenAI API (gpt-4-o)
Command-R	CohereForAI/c4ai-command-r-v01
Llama3-8b	meta-llama/Meta-Llama-3-8B-Instruct

Table 5: List the versions of the models proposed in this work. We used the configurations described in §3 in the repositories for each model *(access verified on 10.09.2025).

B Difference between High- and Low-resource Languages

We define the differences between high-resource (HR) and low-resource (LR) settings using the considerations already taken in previous work (Ranaldi et al., 2024, 2025b). Table 6 reports the language distribution of CommonCrawl, and Table 7 the number of documents in the Wikipedia dump used in our work (§3).

Language	Percentage
English (en)	46.3%
Russian (ru)	6.0%
German (de)	5.4%
Chinese (zh)	5.3%
French (fr)	4.4%
Japanese (ja)	4.3%
Spanish (es)	4.2%
Other	23.1%

Table 6: Language distribution of (Common Crawl, 2021).

C Documents in Wikimedia Dump

Language	Percentage
English (en)	41,488k
Russian (ru)	13,784k
German (de)	20,772k
Chinese (zh)	7,875k
Italian (it)	10,462k
French (fr)	17,813k
Japanese (ja)	6,626k
Spanish (es)	12,865k
Portuguese (pt)	5,637k
Bengali (bn)	767k
Finnish (fn)	272k
Arabic (ar)	1,050k
Thai (th)	876k
Vietnamese (vi)	2,067k
Telogu (te)	124k

Table 7: Language distribution of Wikimedia Dump introduced in §3.

D Retrieval Details

We use Cohere as the retrieval system and Wikimedia_dump as the database. Cohere in *wikipedia-2023-11-embed-multilingual-v3* (available on huggingface) provides individual documents embedded with multilingual embedding model *Cohere_Embed_V3*. For each question in the evaluation data, we retrieve 50 relevant documents and then rerank the top-5 most relevant ones using dot score between query embedding and document embeddings. We use this procedure as recommended in the use cases (case1, case2).

E Retrieval Bergen

To reproduce the retrieval setting proposed in (Chirkova et al., 2024), we used the open-source library available at the following link. We reproduce the same settings operating via BGE-m3.

F Data Composition

As introduced in §3.1 we use *(i)* MLQA (Lewis et al., 2020a), *(ii)* MKQA (Longpre et al., 2021) and *(iii)* XOR-TyDi QA (Asai et al., 2021) as they best represent multilingual open-ended question-answering tasks. MLQA is manually translated from SQuAD v1.1 (Rajpurkar et al., 2016), MKQA and XOR-TyDi QA are machine translated and manually controlled by Natural Questions (Kwiatkowski et al., 2019) and TyDi QA (Clark et al., 2020), respectively. We use parts of the datasets in the languages listed in Table 11. For each language, we used the same questions and, consequently, the same number of questions to avoid any imbalance in double-checking by retrieving the corresponding ids. Details on the number of instances are given in Table 8.

Dataset	#language available	#language used	#Total used
MLQA	$1.5k$	$0.8k$	$9.6k$
MKQA	$2k$	$1.2k$	$8.4k$
XOR-TyDi QA	$0.6k$	$0.4k$	$2.4k$

Table 8: Number of instances used in our evaluations equally distributed among the languages in Table 11. We denote by k 1000 instances.

J Robustness Analysis

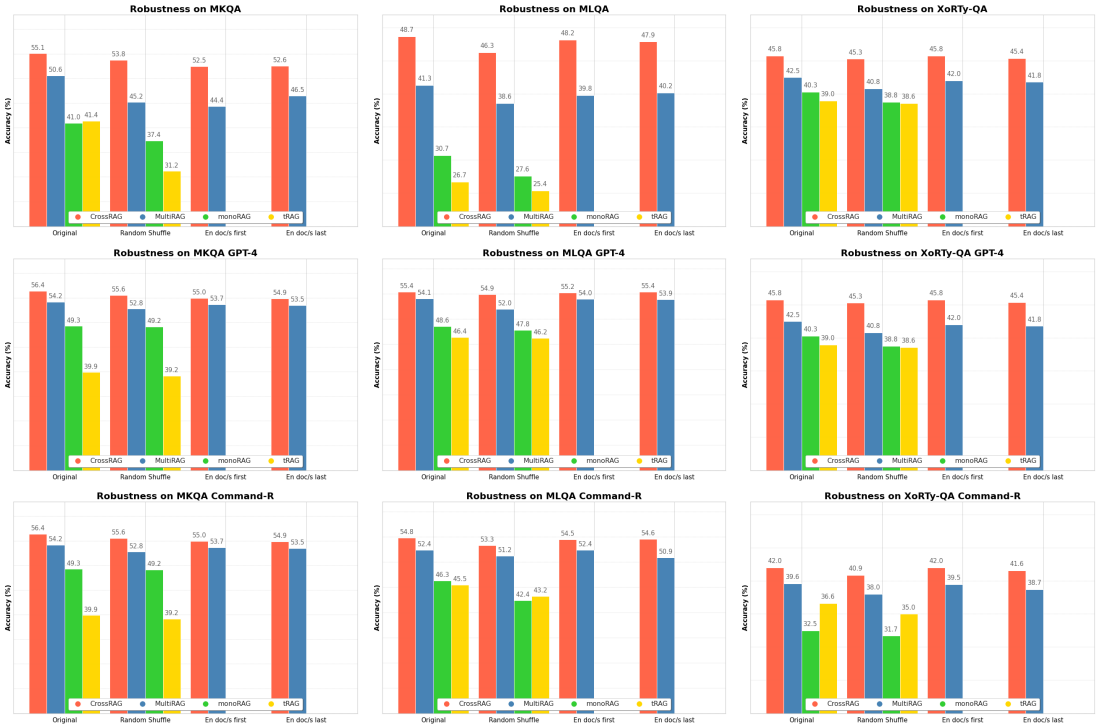


Figure 7: Robustness analysis. We deliver the retrieved documents using the order presented in §3.2 (Original), randomly (Random Shuffle), with English first (En doc/s first) and English last (En doc/s last).

G Translation-Following

We instruct Llama-3-8b using instruction set composed from **Instruction**: `'Translate the following text from` L_X `to English'`, **Input:** `'Sentence in` L_X`'`. and **Output:** `'Sentence in English'.`, as in (Ranaldi et al., 2024). We used *news_commentary* (Tiedemann, 2012) by selecting En-X translations as detailed in Table 9. We randomly extracted 2k demonstrations for the available languages in the open repo (link1, link2). We tune the model for three epochs with a batch size of 32 and a learning rate equal to 1e-5 with a 0.001 weight decay. We use the cosine learning rate scheduler with a warmup ratio of 0.03. We conducted our experiments on a workstation with four Nvidia RTX A6000 48VRAM for approximately 14 GPU/h.

Languages
German, Spanish, Italian, Russian, Chinese, Japanese, Arabic, Russian, Hindi
Total: 18k

Table 9: Instances and languages used for conducting Translation-following experiment.

H Translation-Following Results

To make the experimental setting fair and consistent, we randomly extracted the same number of demonstrations for the languages available in the one-source repositories. Although these offer a large number of languages, some languages (Korean, Finnish, Thai and Vietnamese) are not available. However, we chose not to exclude these languages from the final evaluation.

Language	**MLQA**	**MKQA**
German	69.8	67.5
Italian	69.2	-
Chinese	64.4	62.3
Japanese	56.8	-
Spanish	69.2	69.6
Portuguese	69.0	-
Russian	64.2	-
Arabic	59.1	43.9
Hindi	-	40.6
Finnish	57.0	-
Korean	39.4	-
Thai	23.9	-
Vietnamese	-	44.5
Avg	58.9	54.7
Avg LR	45.2	43.3

Table 10: Performances Llama-3-8b with Translation-following tuning.

I Proposed Task

Dataset	Task	Languages	#Languages
MKQA	QA	English, Spanish, German, Italian, Portuguese, Russian, Chinese, Korean, Thai, Japanese, Finnish, Arabic	12
MLQA	QA	English, Chinese, Arabic, German, Spanish, Vietnamese, Hindi,	7
XORTyDi	QA	English, Chinese, Arabic, Chinese, Korean, Finnish, Telogu, Bengali	8

Table 11: Languages present in datasets used in this work. *We denote question-answering task as (QA)

K Instruction Template

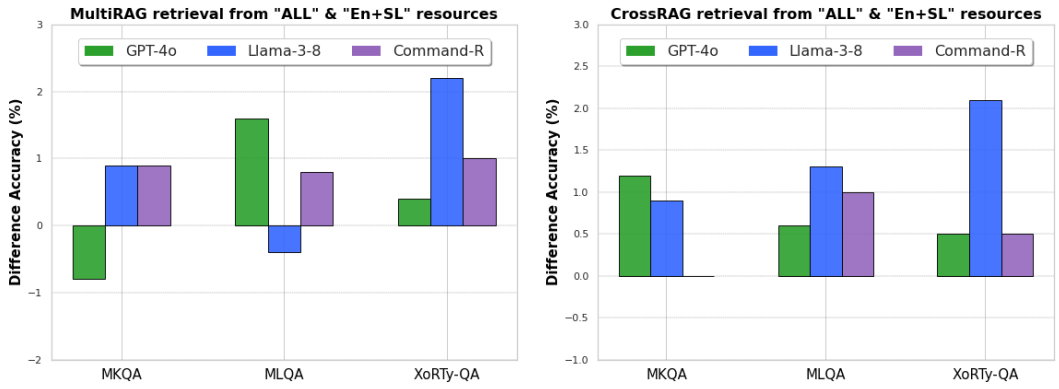
This section contains the *Instruction Templates* used for the additional analysis.

Translation

Please answer the question by following the provided instructions.
#Instructions:
Provide the English translation for this document. Your language and style should align with the language conventions of a native speaker.
 # Document: *[document]*

Table 12: *Instruction Templates*. The structure is defined by a set of in-context examples (zero examples, in the 0-shot case), the question in {**evaluated language**}, the final instruction part and a special template to guide generation and support the final evaluation.

M Performance differences between Retrieval Scope



(a) MRAG Differences (ALL vs En+SL)

(b) CRAG Differences (ALL vs En+SL)

Figure 8: Difference in MRAG (a) and CRAG (b) between retrieval from documents available in Wikidump (ALL) and retrieval done in English+Query specific language (En+SL).

L Languages of Retrieved Documents

		Retrieval from $\mathbf{W}_{En+SL}$ ($\mathcal{R}$ from En and SL Docs)		Retrieval from $\mathbf{W}_{ALL}$ ($\mathcal{R}$ from All Available Docs (ALL))		
	Question Language	**% En**	**% SL**	**% En**	**% SL**	**% Others**
MKQA	English	-	-	98.9%	-	1.1%
	German	10.8%	89.2%	10.2%	86.3%	3.1%
	Italian	12.6%	87.4%	11.8%	85.8%	2.4%
	Spanish	12.4%	87.6%	11.4%	86.0%	2.8%
	Finnish	**26.3%**	73.7%	22.6%	67.1%	10.3%
	Portuguese	12.0%	88.0%	11.7%	85.8%	2.9%
	Russian	**25.3%**	74.7%	22.2%	65.2%	12.6%
	Chinese	16.3%	83.7%	14.4%	81.2%	4.4%
	Arabic	**28.2%**	71.8%	24.3%	66.2%	9.5%
	Japanese	18.2%	81.8%	16.3%	80.3%	5.4%
	Korean	**30.0%**	70.0%	24.0%	65.5%	10.5%
	Thai	**33.3%**	66.7%	26.2%	64.6%	9.2%
MLQA	English	-	-	99.2%	-	0.8%
	Chinese	18.4%	82.6%	15.3%	83.5%	2.2%
	Arabic	**28.1%**	71.9%	20.8%	70.0%	9.2%
	German	14.4%	85.6%	13.0%	85.5%	1.5%
	Spanish	10.7%	89.3%	11.4%	86.0%	2.8%
	Vietnamese	**39.0%**	61.0%	32.2%	55.4%	12.4%
	Hindi	**38.5%**	61.5%	32.6%	58.8%	9.2%
XORTyDi QA	English	-	-	98.4%	-	1.6%
	Arabic	18.4%	81.6%	16.3%	76.6%	7.1%
	Bengali	43.8%	56.2%	40.6%	46.6%	12.8%
	Chinese	16.8%	83.2%	15.6%	79.0%	7.4%
	Korean	34.3%	65.7%	31.2%	59.2%	9.8%
	Russian	23.6%	76.4%	19.8%	68.4%	11.8%
	Finnish	20.6%	79.4%	19.8%	70.8%	9.4%
	Telogu	45.6%	54.4%	42.0%	45.6%	12.4%

Table 13: Percentage of the languages of retrieved documents. We retrieve the documents using $\mathcal{R}$ system from the Wikipedia dump (detailed in §3) considering both English+Specific Language ($\mathbf{W}_{En+SL}$) and all languages analysed in the task ($\mathbf{W}_{ALL}$). The languages are checked using OpenLID framework (Burchell et al., 2023).

N Generated Languages

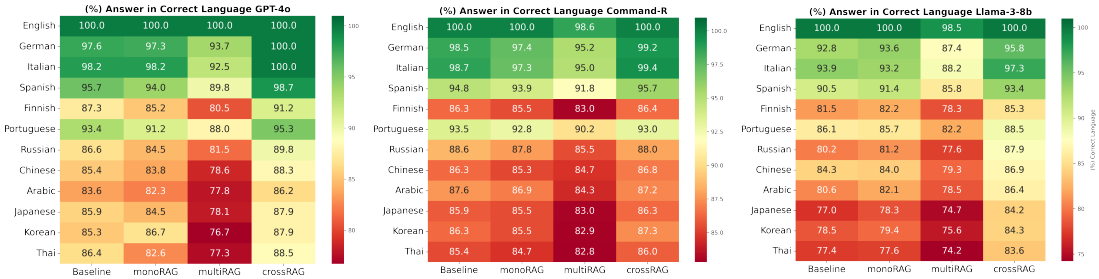


Figure 9: Generated languages from models used in our work (§3.2) on MKQA. The languages are checked using OpenLID framework (Burchell et al., 2023).

O Perfomances using character 3-gram

Model	MKQA			MLQA			XoRTy-QA		
	Avg	HR	LR	Avg	HR	LR	Avg	HR	LR
GPT-4-o	38.3	55.5	30.3	41.2	50.2	30.1	29.8	35.2	27.4
tRAG	50.2	61.1	38.3	50.4	58.5	39.5	38.6	41.3	40.5
monoRAG	49.5	60.5	38.0	50.4	60.5	37.7	39.7	42.1	37.6
MultiRAG	55.7	63.7	45.4	55.9	66.4	46.5	42.1	44.4	40.5
CrossRAG	58.0	66.6	47.9	58.6	68.6	48.6	44.7	46.5	43.2
Command-R	40.2	48.7	31.8	40.1	50.5	30.0	31.0	34.5	24.9
tRAG	42.1	53.8	31.0	47.8	59.8	36.1	38.9	40.9	37.5
monoRAG	46.7	54.9	35.1	48.6	60.5	37.1	35.0	43.6	27.1
MultiRAG	57.0	63.9	45.8	54.5	64.3	45.3	42.0	44.3	41.9
CrossRAG	59.3	67.0	51.1	57.2	68.1	46.1	44.4	48.4	44.5
Llama-3-8	38.7	47.4	30.9	39.9	49.2	29.1	28.9	32.2	26.3
tRAG	43.9	52.2	29.5	47.3	58.8	34.8	37.9	39.2	36.3
monoRAG	43.5	54.0	33.5	47.8	59.4	41.6	39.9	41.1	36.5
MultiRAG	53.5	62.1	43.6	52.7	63.8	40.5	44.1	42.9	38.2
CrossRAG	57.4	65.3	51.0	56.2	66.9	45.8	45.7	46.8	44.2

Table 14: Performance (*character 3-gram recall* as in (Chirkova et al., 2024)) using RAG approaches described in §2 across benchmarks and settings detailed in §3, separated by total average (Avg), high-resource (HR) and low-resource (LR) languages averages.

P Example of bad retrieval in tRAG

Original Question: *¿quién escribió variaciones de Campanita del lugar?*
Translated Question: *Who wrote variations of Tinkerbell of the Place?* *translation by Google API
Target: *[Wolfgang Amadeus Mozart, Mozart]*

[1]: Tinker Bell is a fictional character from J. M. Barrie's 1904 play Peter Pan and his 1911 novelisation *Peter and Wendy*. She has appeared in a variety of film and television adaptations of the Peter Pan stories, notably Walt Disney's 1953 animated film *Peter Pan* and its 2023 live-action adaptation *Peter Pan & Wendy*.
[2]: 'Jingle Bells' is one of the best known and most sung traditional winter songs in the world. It was written between 1850 and 1857 by the American composer James Pierpont (1822– 893) under the title 'The One Horse Open Sleigh' and was published in Boston by Oliver Ditson & Co. on 16 September 1857.
[3]: Tinker Bell is a 2008 American animated film and the first installment in the Disney Fairies franchise produced by DisneyToon Studios. It is about Tinker Bell, a fairy character created by J. M. Barrie in his 1904 play *Peter Pan, or The Boy Who Wouldn't Grow Up*, and featured in the 1953 Disney animated film, *Peter Pan* and its 2002 sequel *Return to Neverland*. Unlike Disney's two Peter Pan films featuring the character, which were produced primarily using traditional animation, Tinker Bell was produced using digital 3D modeling. The film was released on DVD and Blu-ray by Walt Disney Studios Home Entertainment on October 28, 2008. A sequel, *Tinker Bell and the Lost Treasure*, was released in 2009.
[4]: Tinker Bell is an animated fantasy film series based on the character Tinker Bell, produced by DisneyToon Studios as part of the Disney Fairies franchise after producing a number of direct-to-video Winnie the Pooh films. Voices of Mae Whitman, Lucy Liu, Raven-Symoné, America Ferrera, Kristin Chenoweth and Pamela Adlon are featured in the main films. Six feature films and one TV special were produced: *Tinker Bell*, *Tinker Bell and the Lost Treasure*, *Tinker Bell and the Great Fairy Rescue*, *Secret of the Wings*, *Pixie Hollow Games* (the special), *The Pirate Fairy*, and *Tinker Bell and the Legend of the NeverBeast*. The series is a spin-off of and prequel to *Peter Pan*. Originally developed as a direct-to-video franchise, the series was theatrically released from its third film onwards.
[5]: Sir James Matthew Barrie, 1st Baronet, was a novelist and playwright, best remembered as the creator of *Peter Pan*. He was born and educated in Scotland and then moved to London, where he wrote several successful novels and plays.

Table 15: An example of bad document retrieval using **tRAG** (query translated before retrieval). The retrieved documents are relevant to the target query.

Q Example of `MultiRAG`

Question: 英格兰有多少位女王 *(how many queens have been in England?)*
Target: 8

Answer Llama3-8 (no-RAG):
实际统治英格兰的女王有四位：玛格丽特一世，玛丽一世，伊丽莎白一世和伊丽莎白二世。女王人数为 4。
(Four queens actually ruled England: Margaret I, Mary I, Elizabeth I and Elizabeth II. The number of queens is 4.)

Retrieval from from $\mathbf{W}_{ALL}$ (Wikipedia in **ALL** languages):
[1]: Since 1 May 1707, when the Kingdom of England and the Kingdom of Scotland were united to form the Kingdom of Great Britain, there have been 13 monarchs of the United Kingdom. The Kingdoms of England and Scotland had been under a commonwealth confederation under the Stuarts since 24 March 1603, and were formally united on 1 May 1707 to form the Kingdom of Great Britain, which was united with the Kingdom of Ireland on 1 January 1801 to form the United Kingdom of Great Britain and Ireland, and the Irish Free State was formed and became independent [...] (*from* $\boldsymbol{W}_{Zh}$)
[2]: This entry lists all female monarchs in the world, including emperors, queens, princes, grand dukes, duchesses, goddesses and other monarchs, and excludes female royals who have ruled on the throne or as regents. In addition, Elizabeth II, who was monarch of the United Kingdom from 1952 to 2022, ruled over 32 Commonwealth realms [...]. (*from* $\boldsymbol{W}_{En}$)
[3]: In less than 200 years, eight queens ruled the Kingdom of England. Among these queens were Matilda, Lady Jane Grey, Mary I, Elizabeth I, Mary II, Anne, Victoria, and Elizabeth II. Despite the tradition that heirs of the throne should be the eldest son of the monarch, the mentioned queens ruled competence and capability [...] (*from* $\boldsymbol{W}_{En}$)
[4]: Empress Matilda, also known as Empress Maud, was one of the claimants to the English throne during the civil war known as the Anarchy. The daughter and heir of Henry I, king of England and ruler of Normandy, she went to Germany as a child when she was married to the future Holy Roman Emperor Henry V. She travelled with the emperor to Italy in 1116, was controversially crowned empress in St Peter's Basilica [...] (*from* $\boldsymbol{W}_{Zh}$)
[5]: Elizabeth II, known as Elizabeth Alexandra Mary, was Queen of the United Kingdom and other Commonwealth realms from 1952 to 2022 and held the record for the longest reign [...] (*from* $\boldsymbol{W}_{Zh}$)

Answer Llama3-8b (`MultiRAG`):
英格兰曾有8位女王作为君主执政....答案是：8
Thus, England has had 8 queens who ruled as sovereigns.... **The answer is: 8.**

Table 16: An example of **`MultiRAG`** inference for Llama3-8b. To facilitate the reading, we translated retrieved from Chinese Wikipedia ($\boldsymbol{W}_{Zh}$) into English (at inference time, we delivered them in Chinese).

R Example of `MultiRAG` Misleading Answer

Question: 누가 '나는 바비걸' 노래를 만들었나요? *(Who made the song i'm a barbie girl?)*
Target: ['아쿠아' , 'Aqua']

Answer Llama3-8 (no-RAG):
정답은 Ingemar Örhagen, Matthias Lindblom, 그리고 Candice Posey입니다.
(The answer is Ingemar Örhagen, Matthias Lindblom and Candice Posey.)

Retrieval from from $\mathbf{W}_{ALL}$ (Wikipedia in **ALL** languages):
[1]: The music video for "Barbie Girl" was recorded on August 2005, between 17 and 18. It was directed by Ricardo Vereza, Bidu Madio, Rentz and Mauricio Eça. The video was released on August 30. Kelly plays a determinate and feminist woman, who doesn't need a man [...] (*from $\boldsymbol{W}_{Ko}$*)
[2]: Barbie Girl is a song by the Danish-Norwegian dance-pop group Aqua. It was released in May 1997 as the third single from the group's debut studio album, "Aquarium" (1997). The song was written by Søren Rasted, Claus Norreen, René Dif, and Lene Nystrøm, and was produced by Johnny Jam, Delgado, Rasted, and Norreen. It was written after Rasted [...] (*from $\boldsymbol{W}_{En}$*)
[3]: Barbie and The Three Musketeers is a video premiere animated feature from Universal Pictures, released on DVD on 15 September 2009 in the United States. It is the 16th film in the Barbie animated series and is based on the novel Les Trois Mousquetaires by French writer Alexandre Dumas. The heroine of the film is played by Barbie and her three friends, Teresa, Summer and Nikki, who together realise their dream of becoming [...] (*from $\boldsymbol{W}_{Ko}$*)
[4]: Barbie as the Princess and the Pauper is a video premiere animated production released by Lionsgate Entertainment on 28 September 2004 on DVD and VHS. It is the fourth film in the Barbie animated series, based on Mark Twain's classic novel The Beggar Prince. The film is also the first animated musical of the Barbie film series, with a total of seven original songs performed by Barbie in two roles a noble princess and a poor village girl. [...] (*from $\boldsymbol{W}_{Ko}$*)
[5]: The song, at first glance, appears to be about the popular children's doll Barbie. However, at second glance, the song contains several sexual overtones, such as "You can brush my hair, undress me everywhere". This caused some controversy upon its release in Denmark, but despite the controversy and perhaps partly due [...] (*from $\boldsymbol{W}_{En}$*)

Answer Llama3-8b (`MultiRAG`):
The seven original songs in Barbie as the Princess and the Pauper are less certain, as they were likely adapted from classical compositions and reworked into the animated film's soundtrack by an unspecified creative team. **The answer is:** The songs were all composed by Wolfgang Amadeus Mozart in collaboration with contemporary pop producers. (*translated from Korean*)
Answer Llama3-8b (`CrossRAG`):
The song "Barbie Girl" was performed by the Danish-Norwegian [...] **The answer is: Aqua.** (*translated from Korean*)
Answer GPT-4o (`MultiRAG`):
The song "Barbie Girl" was written by Søren Rasted, Claus Norreen, René Dif, and Lene Nystrøm, members of the Danish-Norwegian dance-pop group Aqua. **The answer is: Aqua.** (*translated from Korean*)

Table 17: An example of `MultiRAG` inference. We have translated documents and answers into English to facilitate the understanding as in Table 16.

S Performance translating the Answers in English using GPT-4o

Model	MKQA			MLQA			XoRTy-QA		
	Avg	HR	LR	Avg	HR	LR	Avg	HR	LR
GPT-4-o	41.8	59.7	33.9	45.5	53.2	33.9	33.9	39.2	30.5
tRAG	53.0	64.7	41.9	53.8	61.3	43.1	42.4	45.2	44.4
monoRAG	53.0	64.5	41.2	54.1	64.6	40.8	42.8	45.6	40.9
MultiRAG	60.6	68.4	49.6	59.7	70.8	50.8	45.9	49.1	45.5
CrossRAG	58.0	66.6	47.9	58.6	68.6	48.6	44.7	46.5	43.2
Command-R	43.1	51.3	34.8	43.1	53.8	32.5	34.2	37.4	27.9
tRAG	45.2	56.4	33.5	50.5	62.2	38.7	41.7	44.0	40.1
monoRAG	50.4	58.6	38.5	51.9	63.6	41.0	38.7	47.4	30.6
MultiRAG	61.0	67.4	49.4	58.1	68.4	49.6	46.2	48.2	45.9
CrossRAG	62.9	70.6	55.1	61.3	72.5	50.1	47.9	52.7	48.0
Llama-3-8	40.3	49.0	32.6	41.7	50.8	30.9	30.7	34.1	28.0
tRAG	45.7	54.1	31.3	49.2	60.4	36.7	39.7	40.8	37.8
monoRAG	46.9	57.5	36.7	51.1	62.7	44.7	43.2	44.1	39.8
MultiRAG	56.8	65.2	46.8	55.8	67.0	43.7	47.4	46.1	41.4
CrossRAG	60.7	68.5	54.3	59.4	70.0	49.0	48.8	49.9	47.4

Table 18: Performance (translating responses into English and evaluating outputs in English) using RAG approaches described in §2 across benchmarks and settings detailed in §3, categorized by total average (Avg), high-resource (HR), and low-resource (LR) languages.

T Overlap Document Retrieved

Dataset	English	Spanish	German	Russian	Chinese	Finnish	Arabic	Italian	Korean
MKQA monoRAG	94%	85%	84%	83%	87%	82%	82%	89%	86%
MKQA Multi/CrossRAG	92%	82%	86%	84%	86%	82%	84%	88%	85%
MLQA monoRAG	94%	85%	85%	-	77%	-	86%	-	-
MLQA Multi/CrossRAG	95%	86%	84%	-	79%	-	86%	-	-

Table 19: Performance (%) across different languages on MKQA and MLQA datasets using monoRAG and Multi/CrossRAG retrieval systems.

U Translation in Languages beyond English

Language	MultiRAG	CrossRAG (original)	CrossRAG (translation by TL)
English	73.8	74.4	69.8
German	71.4	72.6	72.6
Italian	68.4	71.6	69.0
Spanish	68.6	70.9	68.7
Finnish	60.4	65.3	55.2
Portuguese	69.1	69.8	68.2
Russian	64.3	65.5	51.8
Chinese	57.2	60.8	53.0
Japanese	44.0	49.1	41.8
Korean	41.3	49.6	42.9
Thai	26.3	33.4	30.2

Table 20: Performance comparison by language and retrieval method. Documents are translated to the nearest typological language (indicated as TL) instead of English (experiments on MKQA with GPT-4o).

Language 1	Language 2	Typological Relationship
Italian	Spanish	Both are Romance languages with very similar grammar and vocabulary
Italian	Portuguese	Romance languages with comparable verb conjugation and noun/adjective agreement
Spanish	Portuguese	Closely related Ibero-Romance languages
English	German	Both are West Germanic languages, sharing syntactic structure and some core vocabulary
Japanese	Korean	Both use SOV word order, agglutinative morphology, and topic/subject marking particles
Chinese	Thai	Both are analytic (isolating) languages with tonal systems and SVO word order
Russian	Finnish	Not genealogically related but share areal features (e.g., case-rich systems, free word order)

Table 21: Typological relationships between language pairs.