How Does Knowledge Selection Help Retrieval Augmented Generation?

Xiangci Li[1, 2*] **Jessica Ouyang**[2]
[1] AWS AI Labs, [2] University of Texas at Dallas
lixiangci8@gmail.com, jessica.ouyang@utdallas.edu

Abstract

Retrieval-augmented generation (RAG) is a powerful method for enhancing natural language generation by integrating external knowledge into a model's output. While prior work has demonstrated the importance of improving knowledge retrieval for boosting generation quality, the role of knowledge *selection*, a.k.a. *reranking* or *filtering*, remains less clear. This paper empirically analyzes how knowledge selection influences downstream generation performance in RAG systems. By simulating different retrieval and selection conditions through a controlled mixture of gold and distractor knowledge, we assess the impact of these factors on generation outcomes. Our findings indicate that the downstream generator model's capability, as well as the complexity of the task and dataset, significantly influence the impact of knowledge selection on the overall RAG system performance. In typical scenarios, improving the knowledge recall score is key to enhancing generation outcomes, with the knowledge selector providing limited benefit when a strong generator model is used on clear, well-defined tasks. For weaker generator models or more ambiguous tasks and datasets, the knowledge F1 score becomes a critical factor, and the knowledge selector plays a more prominent role in improving overall performance.

1 Introduction

Retrieval-augmented generation (RAG) has become a pivotal technique in natural language generation, enhancing a language model's ability to produce relevant, informed output by incorporating external knowledge (Gao et al., 2023; Fan et al., 2024; Gan et al., 2025). This approach complements the model's internal knowledge, which is inherently constrained by the information that was available during training, by retrieving up-to-date, relevant information to support and inform its output.

The quality of the retrieved knowledge directly influences the quality of RAG outputs. Previous studies have consistently shown that improving knowledge *retrieval* leads to a direct enhancement in generation performance (Dinan et al., 2019; Li et al., 2022, 2024; Wang et al., 2024; Wu et al., 2024). Similarly, effective knowledge *selection*, a.k.a. *reranking* or *filtering*, has been observed to improve generation quality by filtering out retrieved information that is less relevant to the generation target (Kim et al., 2020; Thulke et al., 2021; Li et al., 2022; Sun et al., 2023; Zhang et al., 2023; Wang et al., 2023; Zheng et al., 2024; Wang et al., 2024; Zhao et al., 2025). However, we observe that knowledge selection is barely used for tasks other than dialogue generation, and there are notably fewer LLM-based RAG works that use knowledge selection, compared to fine-tuned RAG models (see Section 2).

We hypothesize that knowledge selectors may not always improve downstream generation performance, and that there may be a selection bias where only positive results involving knowledge selector modules are published, while experiments where knowledge selectors are not helpful simply do not report those results. Further, prior works focus on proposing specific knowledge selection approaches, offering only narrow, case-specific insights into the relationship between knowledge and generation. As a result, readers are often left with anecdotal observations, such as "using model A in scenario X improves performance", without a global picture of how knowledge selection impacts generation and when it is most effective.

Therefore, in this work, we perform a systematic empirical analysis[*] of how knowledge selectors

[*] Work performed outside of AWS AI Labs.

[*] https://github.com/jacklxc/KnowledgeSelectionSimulation

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 4104–4121
November 4-9, 2025

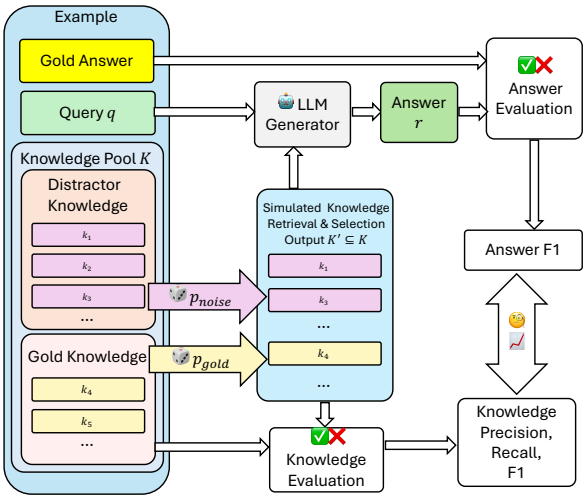

Figure 1: Our simulation experiment pipeline.

with various performances affect downstream RAG performance. By blending gold knowledge with distractor knowledge in varying ratios, we *simulate* different selection outcomes and examine their impact on the overall performance of RAG. We find that the generator model's capability, as well as the complexity of the task and dataset, significantly influence RAG system performance. In typical scenarios, improving knowledge *recall* via the knowledge retriever is key to enhancing generation outcomes, with the knowledge selector providing limited additional benefit when a strong generator model is used on clear, well-defined tasks. For weaker generator models or more ambiguous tasks and datasets, knowledge *F1* becomes a critical factor, and the knowledge selector plays a more prominent role in improving overall performance.

2 Related Work

Retrieval-augmented generation (RAG) has been extensively studied in recent years. Early works (Guu et al., 2020; Lewis et al., 2020b; Shuster et al., 2021) jointly fine-tuned a dense retriever (e.g. DPR (Karpukhin et al., 2020)) and generator (e.g. BART (Lewis et al., 2020a)), which required dedicated training datasets. More recently, the introduction of large language models (LLMs) has made RAG more convenient to implement due to their strong generation performance, in-context learning ability (Brown, 2020; Kojima et al., 2022), and drastically longer context windows — from BART's 1024 tokens to more than a million for modern LLMs (Lee et al., 2024). As a result, recent RAG research has shifted to using LLMs (Gao et al., 2023; Fan et al., 2024; Gan et al., 2025); in this work, we follow this trend to focus on LLM-based RAG.

Knowledge selector for knowledge-grounded dialogue generation. Dialogue generation (Moghe et al., 2018; Dinan et al., 2019; Li et al., 2024) is one of the major applications of RAG, where the target response is conditioned on retrieved knowledge, and a knowledge selection step is often added to further refine the retrieved knowledge (Thulke et al., 2021; Sun et al., 2023; Zhang et al., 2023). For example, Kim et al. (2020) train a knowledge selector by leveraging response information, Li et al. (2022) select knowledge from document semantic graphs, Zhang et al. (2023) propose multi-task learning for knowledge selection and response generation, and Zhao et al. (2025) proposes a multi-step reranking process for Natural Question (Kwiatkowski et al., 2019) and Trivia QA (Joshi et al., 2017). However, while these works demonstrate the advantages of their approaches through ablation studies, it is unclear whether their performance improvement via knowledge selection specifically is transferable to other datasets, domains, or tasks. Moreover, despite the popularity of LLMs for RAG, there are dramatically fewer LLM-based works that include the knowledge selection step, and it is likewise omitted in other RAG tasks (Gao et al., 2023; Fan et al., 2024; Gan et al., 2025).

To our knowledge, there is no prior work explaining this gap in knowledge *selection*. The closest works to ours are Cuconasu et al. (2024), Wu et al. (2024) and Jin et al. (2025), which study the impact of knowledge *retrieval* on RAG. We hypothesize that there is a selection bias effect where knowledge selection modules may have been experimented with, but ultimately not reported, in tasks or settings where it is not helpful. In this paper, we study the general effect of knowledge retrieval and selection on LLM-based RAG via hundreds of simulations, rather than being restricted to a few specific, anecdotal retrievers or selector implementations like prior works.

3 Approach

3.1 Task Formulation

RAG involves three steps: (1) Knowledge retrieval, where a retriever module retrieves a set of candidate knowledge K based on a query q. This step

aims to retrieve as much knowledge that is relevant to the query as possible, balancing knowledge *recall* and *precision*. (2) Optionally, knowledge selection, a.k.a. reranking or filtering, removes retrieved knowledge that is less relevant to further improve knowledge *precision*, producing $K' \subseteq K$. (3) Finally, the generator takes the query q and the selected, retrieved knowledge K' to generate the output text r.

The knowledge in K and K' can be heterogeneous and come from multiple sources, such as external documents, knowledge graphs, or conversation histories. This RAG framework can be applied to various tasks, such as dialogue generation, question answering, fact verification, and code generation (Gao et al., 2023; Fan et al., 2024; Gan et al., 2025). In our experiments on dialogue generation and question answering, we *simulate* steps 1 and 2 by creating controlled knowledge sets K' and observing the resulting generation performance of step 3.

3.2 Knowledge Simulation

We aim to perform a systematic analysis of the effect of knowledge retrieval and selection outcomes on the downstream RAG performance by *simulation*. As Figure 1 shows, for each query q, given a fixed pool of available knowledge with gold relevance annotations, we sample gold knowledge and distractor knowledge at varying rates p_{gold} and p_{noise} to precisely simulate a wide range of quality for the retrieved and selected knowledge K', from noise only to gold knowledge only. For example, if $p_{gold} = 0.5$, then each piece of gold knowledge has a 50% chance to be sampled. Each sampling produces a full experiment over the *entire test set*, reported as one data point in Figures 2-4.

In this way, we simulate a wide distribution of knowledge retriever and selector performance, as measured by knowledge precision and recall based on the gold annotations. We then test the performance of the generator model given the different quality of simulated retrieved and selected knowledge K'. Compared to prior works discussed in Section 2, which only compare a few ablation configurations, we conduct hundreds of configurations in each meta-experiment.

4 Experimental Settings

4.1 Datasets

While there are several datasets supporting the RAG framework, relatively few provide high-quality, human-annotated gold knowledge. Furthermore, the target outputs should be relatively short, unambiguous, and easily evaluated with automatic metrics such as F1 scores. These conditions narrow down the choices to two popular and representative datasets, Wizard of Wikipedia (WoW; Dinan et al., 2019) and HotpotQA (Yang et al., 2018).

WoW is an open-domain dialogue dataset based on Wikipedia knowledge that has been widely used in prior knowledge selection works (Kim et al., 2020; Thulke et al., 2021; Li et al., 2022; Sun et al., 2023). The wizard speaker (a human annotator playing the part of the dialogue system) has access to Wikipedia knowledge, while the apprentice speaker does not. The apprentice is given a starting conversation topic, but otherwise speaks freely. The last two turns of dialogue are used as the query q to *retrieve* Wikipedia passages (K), and the wizard *selects* a single knowledge sentence (K') to generate their response.

Despite its popularity, this dialogue generation task is challenging to evaluate. First, because the wizard is limited to selecting only a single knowledge sentence, it is still possible for the unselected "distractor" knowledge to be relevant to the wizard's response. Second, as is the case in many natural language generation tasks, the gold wizard responses in WoW are not the only plausible responses, making it harder to quantify the correctness of a generated response. Nonetheless, WoW is a well-annotated dialogue dataset that is close to a real-life scenario where the gold knowledge and responses are noisy.

HotpotQA is a question-answering dataset derived from Wikipedia knowledge that we include in our experiments to mitigate the ambiguity of WoW. The multi-hop questions and answers are first directly derived from gold knowledge graphs, and then distractor knowledge is injected to create noise. As a result, unlike WoW, the answers in HotpotQA are strongly dependent on the gold knowledge, and the distractor knowledge is unlikely to be relevant. The short and unambiguous gold answers make it simpler to evaluate via F1 scores.

Input Knowledge	KP	KR	KF1	R-L	F1
GPT-4o-mini					
No knowledge	0	0	0	0.110	0.200 (± .005)
Full knowledge	0.015	1	0.031	0.140	0.251 (± .006)
Gold knowledge	1	1	1	0.167	0.276 (± .007)
LLaMA 3.1 8B					
No knowledge	0	0	0	0.111	0.216 (± 0.005)
Full knowledge	0.015	1	0.031	0.138	0.248 (± .005)
Gold knowledge	1	1	1	0.164	0.278 (± .008)
Mistral 7B Instruct					
No knowledge	0	0	0	0.113	0.203 (± .005)
Full knowledge	0	1	0	0.131	0.233 (± .005)
Gold knowledge	1	1	1	0.172	0.268 (± .007)

Table 1: WoW response generation performance benchmarked by different LLM generators. We measure knowledge precision (KP), recall (KR), and F1 (KF1); response ROUGE-L F1 (R-L); and response F1 (and its standard error mean).

Input Knowledge	KP	KR	KF1	EM	F1
GPT-4o-mini					
No knowledge	0	0	0	0.330	0.437 (± .020)
Full knowledge	0.065	1	0.120	0.668	0.780 (± .016)
Gold knowledge	1	1	1	0.710	0.828 (± .014)
LLaMA 3.1 8B					
No knowledge	0	0	0	0.200	0.298 (± .019)
Full knowledge	0.065	1	0.120	0.372	0.545 (± .019)
Gold knowledge	1	1	1	0.414	0.671 (± .016)
Mistral 7B Instruct					
No knowledge	0	0	0	0.208	0.260 (± .019)
Full knowledge	0.065	1	0.120	0.046	0.151 (± .011)
Gold knowledge	1	1	1	0.502	0.627 (± .019)

Table 2: HotpotQA answer generation performance benchmarked by different LLM generators. We measure knowledge precision (KP), recall (KR), and F1 (KF1); answer exact match (EM); and answer F1 (and its standard error mean).

4.2 Generators

To align with the latest trends in RAG research, as well as to make the analysis simple and generally replicable, we adopt LLMs as our generators. Due to the large computational costs for the meta-experiments, each of which consists of hundreds of full experiments, we use three API-based lightweight LLMs, OpenAI GPT-4o-mini, LLaMA 3.1 8B, and Mistral 7B-Instruct, as our generator models; the varying performances of the LLMs allow us to investigate the impact of generator complexity on knowledge usage.

4.3 Knowledge Sampling

As Figure 1 shows, both WoW and HotpotQA are *knowledge selector training datasets* that provide a retrieved knowledge set K for each query q such that the gold knowledge is a subset of K. In both datasets, K contains text passages consisting of an article title and a few knowledge sentences. We perform knowledge sampling at the sentence level to simulate the end result of applying both the knowledge retrieval and selection steps. More details are in Appendix A.1.1.

In our experiments, we refer to using the entire retrieved knowledge set K provided by the dataset as the input knowledge to the generator (i.e. no knowledge selection) as the "full knowledge" setting, where both gold and distractor knowledge are present. We also include a "no knowledge" setting, where the generators receive no external knowledge at all, as a weak baseline, and "gold knowledge," corresponding to perfect knowledge selection, as an upper bound.

5 Meta-Experimental Results

Overall we observe several consistent trends across meta-experiments, except for Mistral-7B-Instruct, which is a relatively weak-performing LLM. Note that each of the plots in Figures 2-4 corresponds to a meta-experiment, and *each point corresponds to a full experiment* on the entire test set. The standard error mean values in Tables 1 & 2 indicate that the answer F1 scores are robust across all experiments and the overall observed trends are significant.

RAG for LLM is beneficial. As Tables 1 & 2 show, generators without any retrieved knowledge ("no knowledge") perform poorly on WoW and HotpotQA, even though they may have been pretrained on the Wikipedia articles that WoW and HotpotQA are derived from. This finding indicates the LLMs are not over-fitted on the WoW and HotpotQA datasets, and applying RAG is beneficial.

Interestingly, our HotpotQA results in Figure 2 show that distractor knowledge significantly harms performance; the cyan dots show that generators receiving mostly distractor knowledge underperform the "no knowledge" setting, which uses only the LLM's internal knowledge. This trend is not observed with WoW, supporting our observation that HotpotQA's distractor knowledge is truly irrelevant, while the "distractor" knowledge in WoW may only be *less* relevant than the single gold sentence, but not completely *ir*relevant. This difference between datasets is further explored in Appendix A.4.

Full knowledge setting without knowledge selection is a strong baseline. Tables 1 & 2 show that the "full knowledge" setting, which corresponds

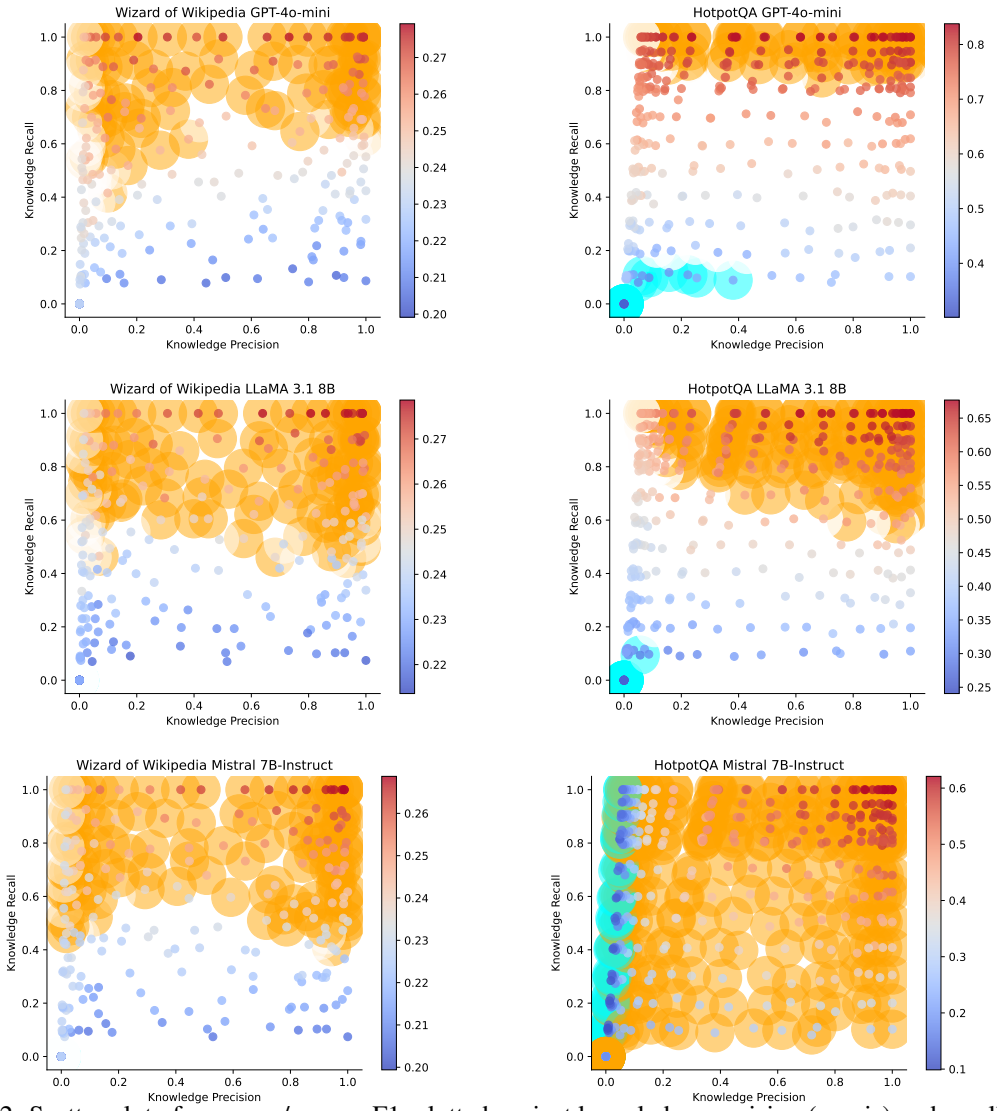

Figure 2: Scatter plot of response/answer F1, plotted against knowledge precision (x-axis) and recall (y-axis), by GPT-4o-mini (top), LLaMA 3.1 8B (middle), and Mistral 7B-Instruct (bottom). The left column shows results on WoW; the right shows HotpotQA. The dots highlighted in orange indicate settings outperforming the "full knowledge" setting, while those highlighted in cyan indicate settings underperforming the "no knowledge" setting. Each figure is a meta-experiment, and each data point corresponds to a full experiment on the entire sampled dataset.

to knowledge retrieval with perfect recall, but no knowledge selection, is a very strong baseline.

This finding mostly contrasts with those of prior knowledge selection works (Kim et al., 2020; Thulke et al., 2021; Li et al., 2022; Sun et al., 2023; Zhang et al., 2023; Zheng et al., 2024; Wang et al., 2024). For example, in Table 2, we find that for GPT-4o-mini, a very strong generator model, the "full knowledge" setting achieves 0.780 answer F1 on HotpotQA, only 0.048 lower than the "gold knowledge" setting; similar observations are seen for LLaMA 3.1 8B on HotpotQA and for both models on WoW. Since the performance gap between the "full knowledge" and "gold knowledge" settings is the space for a knowledge selector to improve generation performance, we conclude that strong generators simply have less room for improvement via knowledge selection.

Knowledge precision & recall together are good predictors of generation performance. Figure 2 shows that generation performance varies smoothly with knowledge precision and recall, indicating that knowledge precision and recall together are major determinants of generation performance. Due to the pipeline nature of the retriever, selector, and generator components of RAG, knowledge *recall* can only be improved by the *retriever*. Meanwhile, the downstream *selector* can only improve knowledge *precision*, but it is likely to also *reduce recall*. Therefore, we can visualize the effect of applying a knowledge selector: it moves the RAG perfor-

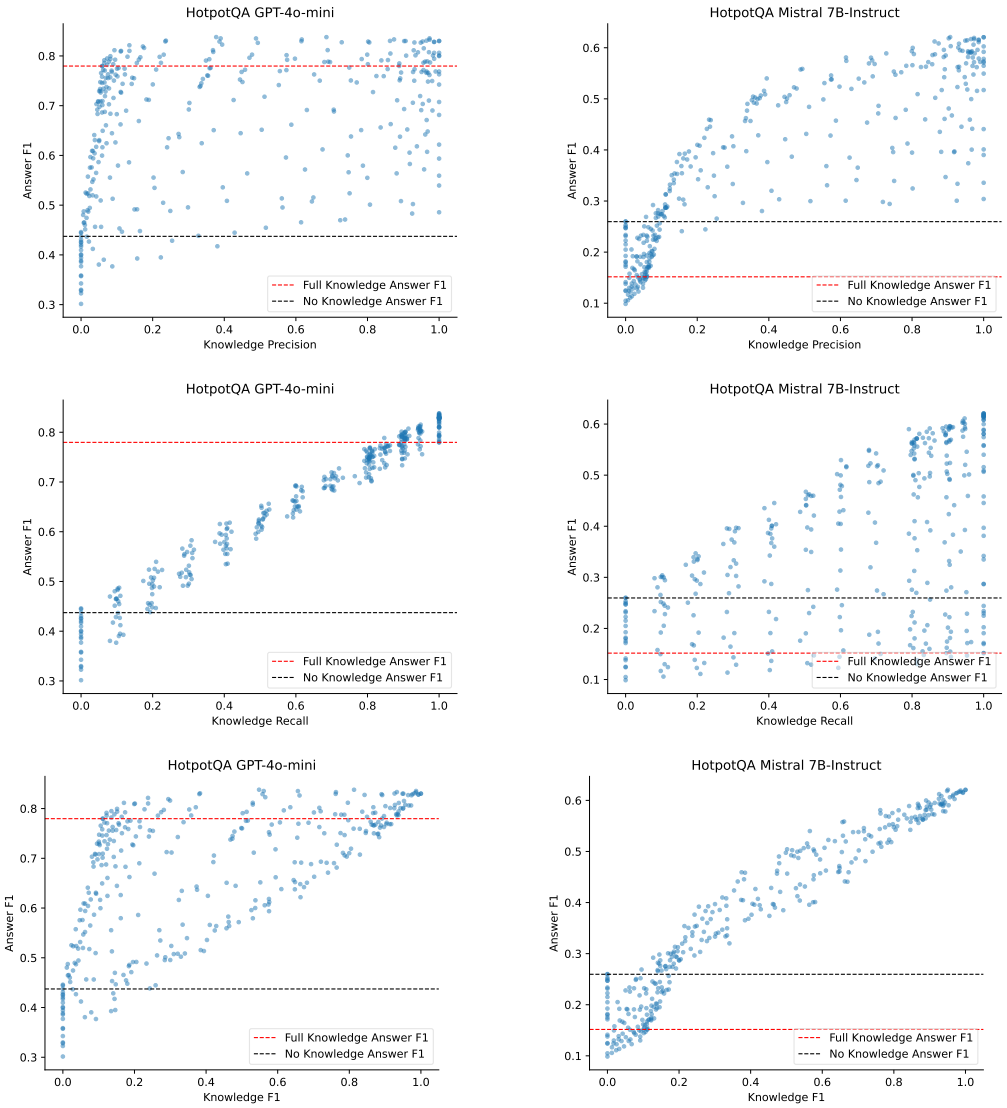

Figure 3: Scatter plot of HotpotQA answer F1 versus knowledge precision (top), knowledge recall (middle), and knowledge F1 (bottom). The left column shows GPT-4o-mini as the generator; the right column shows Mistral-7B-Instruct. Plots for LLaMa 3.1 8B and the WoW dataset are in Appendix A. Each figure is a meta-experiment, and each data point corresponds to a full experiment on the entire sampled dataset.

mance down and to the right in Figure 2; the better the knowledge selector, the more it will move *right* (improving precision), and the less it will move *down* (reducing recall).

Knowledge recall is the most crucial knowledge metric for strong generators. We find that for strong generator models, the knowledge recall score is the best single knowledge metric for estimating generation performance; Figure 3 shows a very strong correlation between knowledge recall and answer F1 for GPT-4o-mini on HotpotQA, and Figure 11 in the Appendix shows the same for LLaMA 3.1 8B on HotpotQA and for both models on WoW. Moreover, knowledge recall is the most important factor in improving generation performance given a fixed generator, while the knowledge precision score is a secondary factor.

Figure 4 further shows that increases in knowledge recall correspond to significant increases in answer F1 scores. Taking GPT-4o-mini on HotpotQA as an example, we see that moving from the left end of a color contour to the right (i.e. keeping knowledge recall fixed while improving precision) only slightly improves answer F1. In contrast, moving from one contour to another (i.e. varying knowledge recall) significantly impacts answer F1 for non-zero precision scores. Thus, to improve knowledge quality for RAG, improving the retriever's recall score is the top priority; improving precision via a knowledge selector has a limited contribution, especially for strong generators.

For weaker generators, like Mistral 7B-Instruct,

the relationship between generation and knowledge F1 is stronger, and correlation with recall is weaker (Figure 3 right). We see much less separation between recall color contours and a much steeper increase with precision (Figure 4 bottom).

Generator capability determines both overall performance and the usefulness of knowledge selection. All of our results show that the overall RAG performance depends on the generator model: the cross-model comparisons in Figures 2 & 4 show that for stronger generator models (as measured by rank in LLM leaderboards[*]), performance across all knowledge settings are higher (Tables 1 & 2). While this result is unsurprising, we also find that the gap between the "full knowledge" and "gold knowledge" settings becomes narrower, suggesting that stronger generators are more robust to noisy input knowledge and rely less on the knowledge selector. In contrast, when the generator is weaker, any reasonable knowledge selector becomes beneficial, likely because a weak generator cannot handle noisy input and requires a selector to filter out distractor knowledge; knowledge F1 best correlates with answer F1 for Mistral-7B-Instruct in Figure 3.

Task and dataset are also key factors in RAG performance. Figures 2 & 4 show that the same generator can show drastically different performance trends between WoW and HotpotQA. For example, while Mistral-7B-Instruct's performance degrades without a knowledge selector on HotpotQA, this is not the case for WoW.

In addition, Figure 4 shows that attempting to improve the knowledge selector on top of a weak retriever (i.e. increasing precision when recall is low) *hurts* generation performance on WoW, as well as GPT-4o-mini on Hotpot QA. In this scenario, more total knowledge, regardless of noise, improves response generation. These counter-intuitive observations are likely due to the nature of the task and the annotation quality. The solution space in HotpotQA is small given a specific question, and HotpotQA has a cleaner separation between gold and distractor knowledge, whereas in WoW, there are more plausible responses given the same conversation history and knowledge, as well as "distractor" knowledge that may actually be relevant to the gold response (see Appendix A.4 for more analysis).

[*] https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard

Non-monotonic trends in improving the knowledge selector. Interestingly, we observe that the boundary of where the knowledge selector improves generation performance (the border between the orange and white areas in Figure 2) is convex for all three LLMs on WoW. This phenomenon can also be seen in Figure 4, where some color contours intersect with the "full knowledge" baseline (dashed red line) multiple times. In other words, we observe that generation performance is non-monotonic as knowledge precision increases with a fixed knowledge recall. The fact that this phenomenon is only observed on WoW may be due to the relatively noisy gold knowledge annotations in WoW; we can produce similar behavior by artificially injecting noise into HotpotQA's gold knowledge annotations (Appendix A.5).

Constraining the knowledge size does not change generation accuracy. One important motivation for using a knowledge selector is to reduce the total input length to the generator model. However, while computational costs can be reduced by only using the top-k knowledge, we find that the overall relationship between knowledge precision-recall and generation F1 observed in our simulations remains unchanged, regardless of the value of k (Appendix A.6).

6 Conclusion and Discussion

In this study, we systematically examined the behavior of RAG generation in relation to the performance of knowledge retrieval and selection.

6.1 Overall Observations

Summing up Section 5, we find an interaction effect between generator capability and ambiguity of the task/dataset on RAG generation performance. We find two types of generator behavior:

A strong generator model can achieve good performance without a knowledge selector because it is robust to noise, and therefore performs better as more gold knowledge is retrieved, despite the presence of distractor knowledge. Thus, knowledge recall correlates well with generation F1, and there is less room for improvement through the addition of a knowledge selector.

A weak generator model cannot handle distractor knowledge and requires a selector to refine the noisy input knowledge, and thus knowledge F1 correlates well with generation F1. Since nowadays most popular generator models are strong LLMs,

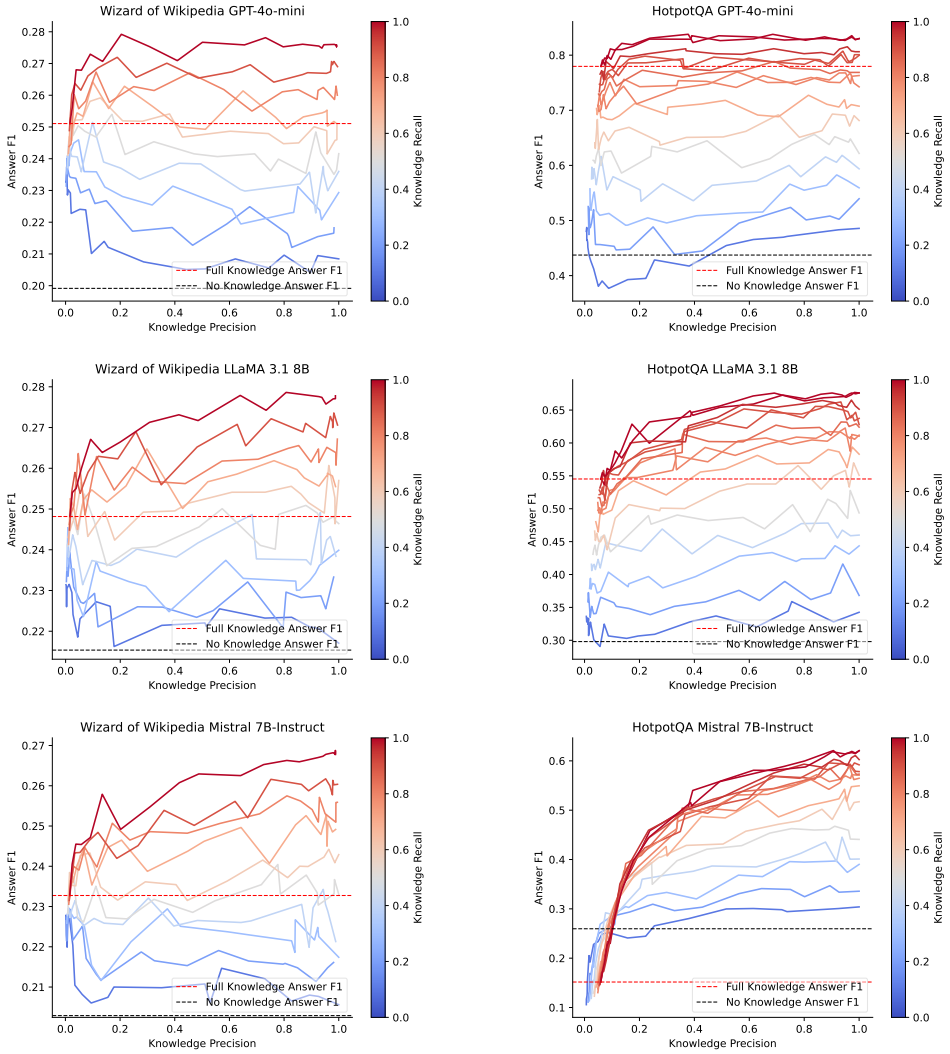

Figure 4: Color contours of answer F1 versus knowledge precision for GPT-4o-mini (top), LLaMA 3.1 8B (middle), and Mistral 7B-Instruct (bottom); the left column shows results on WoW, and the right shows HotpotQA. Each contour represents a different knowledge recall score; moving left to right visualizes improving the performance (precision) of the knowledge selector. Each plot point corresponds to a full experiment on the entire sampled dataset.

knowledge selectors have a limited benefit. We hypothesize that prior work that found performance improvements from dedicated knowledge selectors saw those benefits because their generator models were weak; most prior work used BART (Lewis et al., 2020a) as the generator.

Finally, we note that the strength of a generator model is relative; even a SOTA generator can fall into the "weak" category, given a sufficiently noisy and challenging task/dataset. We hope the visualizations shown in our figures can help guide future practitioners improve their RAG systems in real-world applications.

6.2 Recommendations to Practitioners

Based on our observations, we make the following recommendations for future practitioners considering using knowledge selectors to improve performance in a real-world RAG scenario:

Benchmark the generation performance without external knowledge, with all candidate knowledge, and if possible, with gold knowledge only. The "no knowledge" setting measures the generator's base performance, "full knowledge" serves as a strong baseline corresponding to knowledge retrieval with perfect recall, and "gold knowledge" gives the upper bound of RAG performance with perfect knowledge selection. The gap between "full knowledge" and "gold knowledge" is the potential performance gain brought by adding a knowledge selector.

To improve RAG performance, increasing the knowledge recall score is the most effective strategy. Thus, for modern LLM generators, improving the knowledge retriever's recall is key. In practice, the retriever or selector may encounter false-negative gold knowledge during training, as is the

case with WoW, which makes prioritizing recall even more important. Moreover, because modern LLM generators have long maximum input windows (e.g. 128k for GPT-4o & -mini), having too much knowledge is less likely to be a problem. If the number of knowledge sentences must be constrained to reduce computational cost, the top-k knowledge sentences should maintain the same level of knowledge recall and precision as the non-length-constrained settings.

Only when the knowledge recall score is high can a knowledge selector, which increases knowledge precision, be potentially helpful[*]. Moreover, if the recall is too low, the downstream generation performance may not see any benefits from a knowledge selector with middling performance — it may even be harmed; only when the knowledge selector's precision is very high will the overall performance improve.

Limitations

Computational resources for simulations. Due to the high cost of larger-scale simulations in each meta-experiment that consists of hundreds of full experiments using API-based LLMs, we use only a subset of the WoW and HotpotQA for experiments. As a result, our data subset may contain some minor noise during knowledge sampling and LLM generation, and the contours in Figure 4 are not smooth. However, such noise is not likely to affect our conclusions. For a similar reason, we only chose three LLMs for our experiments. As a result, we may have missed out on more subtle phenomena that can only be seen from results on a larger number of generators.

Datasets for in-depth analysis. In addition to the cost issue, there are very few RAG datasets with human-annotated knowledge (Friel et al., 2024), which further limits our simulation experiment settings. Furthermore, even though we regard WoW as a relatively noisy dataset in this work compared to HotpotQA, to the best of our knowledge, WoW is one of the most cleanly annotated datasets among all datasets. We cannot verify our hypothesis in Section 5 that WoW has a larger solution space than HotpotQA without re-annotating the dataset because each example in WoW only contains one gold response. Our experiments in Appendix A.4 compare the noisiness of gold knowledge annotations in WoW with those in HotpotQA.

Uniform Sampling Probably for Simulation. In Section 3.2, we draw gold and distractor knowledge from a uniform distribution p_{gold} and p_{noise} for simplicity since we do not have a prior assumption of the knowledge selector's preference. However, a real knowledge selector may be more likely to select one knowledge sentence than another. Nonetheless, we successfully identified the knowledge precision and recall scores together as good predictors of the generation performance as we analyzed in Section 5.

Ethics Statement

Since all datasets and LLMs used in this work are publicly available or accessible, and no data collection introducing sensitive information is performed, the ethical considerations of this study are minimal.

[*]In a length-constrained scenario, if we consider k randomly sampled knowledge sentences as a baseline, rather than the "full knowledge" setting, then the knowledge selector is more likely to be helpful due to weaker baseline performance.

References

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–729.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

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. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Robert Friel, Masha Belyi, and Atindriyo Sanyal. 2024. Ragbench: Explainable benchmark for retrieval-augmented generation systems. *arXiv preprint arXiv:2407.11005*.

Aoran Gan, Hao Yu, Kai Zhang, Qi Liu, Wenyu Yan, Zhenya Huang, Shiwei Tong, and Guoping Hu. 2025. Retrieval augmented generation evaluation in the era of large language models: A comprehensive survey. *arXiv preprint arXiv:2504.14891*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. 2025. Long-context LLMs meet RAG: Overcoming challenges for long inputs in RAG. In *The Thirteenth International Conference on Learning Representations*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Byeongchang Kim, Jaewoo Ahn, and Gunhee Kim. 2020. Sequential latent knowledge selection for knowledge-grounded dialogue. In *International Conference on Learning Representations*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

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.

Jinhyuk Lee, Anthony Chen, Zhuyun Dai, Dheeru Dua, Devendra Singh Sachan, Michael Boratko, Yi Luan, Sébastien MR Arnold, Vincent Perot, Siddharth Dalmia, et al. 2024. Can long-context language models subsume retrieval, rag, sql, and more? *arXiv preprint arXiv:2406.13121*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 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.

Sha Li, Mahdi Namazifar, Di Jin, Mohit Bansal, Heng Ji, Yang Liu, and Dilek Hakkani-Tur. 2022. Enhancing knowledge selection for grounded dialogues via document semantic graphs. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2810–2823, Seattle, United States. Association for Computational Linguistics.

Xiangci Li, Linfeng Song, Lifeng Jin, Haitao Mi, Jessica Ouyang, and Dong Yu. 2024. A knowledge plug-and-play test bed for open-domain dialogue generation. In *Proceedings of the 2024 Joint International*

Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 666–676, Torino, Italia. ELRA and ICCL.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2322–2332, Brussels, Belgium. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Weiwei Sun, Pengjie Ren, and Zhaochun Ren. 2023. Generative knowledge selection for knowledge-grounded dialogues. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2077–2088, Dubrovnik, Croatia. Association for Computational Linguistics.

David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2021. Efficient Retrieval Augmented Generation from Unstructured Knowledge for Task-Oriented Dialog. In *Workshop on DSTC9, AAAI*.

Dingmin Wang, Qiuyuan Huang, Matthew Jackson, and Jianfeng Gao. 2024. Retrieve what you need: A mutual learning framework for open-domain question answering. *Transactions of the Association for Computational Linguistics*, 12:247–263.

Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. 2023. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Siye Wu, Jian Xie, Jiangjie Chen, Tinghui Zhu, Kai Zhang, and Yanghua Xiao. 2024. How easily do irrelevant inputs skew the responses of large language models? In *First Conference on Language Modeling*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Yeqin Zhang, Haomin Fu, Cheng Fu, Haiyang Yu, Yongbin Li, and Cam-Tu Nguyen. 2023. Coarse-to-fine knowledge selection for document grounded dialogs. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Xinping Zhao, Yan Zhong, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Dongfang Li, Baotian Hu, and Min Zhang. 2025. FunnelRAG: A coarse-to-fine progressive retrieval paradigm for RAG. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3029–3046, Albuquerque, New Mexico. Association for Computational Linguistics.

Xinxin Zheng, Feihu Che, Jinyang Wu, Shuai Zhang, Shuai Nie, Kang Liu, and Jianhua Tao. 2024. Ks-llm: Knowledge selection of large language models with evidence document for question answering. *arXiv preprint arXiv:2404.15660*.

A Appendix

A.1 Implementation Details

A.1.1 Knowledge Sampling

Since knowledge precision and recall are the most common metrics for knowledge retrieval and selection performance, we use these metrics as the basis for our analysis. We find that the sampling rate p_{gold} linearly correlates with knowledge recall scores, while p_{noise} exponentially correlates with knowledge precision. Thus, to simulate retrieving and selecting a knowledge set K' with a specific knowledge precision and recall score, we use grid search in the linear space of p_{gold} and both the linear and exponential spaces of p_{noise} to ensure most grids in the knowledge precision-recall space are covered by our experiments.

We maintain the original order of the knowledge sentences from the documents provided in the datasets and do not observe a strong influence from the position of the gold knowledge sentences.

A.1.2 Models

We do not perform any fine-tuning or hyperparameter tuning. We set the LLMs' temperature to 0 and ensure that our zero-shot generation prompts (Appendix A.2) are shorter than their maximum input lengths. We use the API services of OpenAI gpt-4o-mini-2024-07-18[*], Together AI[*] meta-llama/Meta-Llama-3.1-8B-Instruct-Turbo, and mistralai/Mistral-7B-Instruct-v0.1. We report results on the first 500 examples in the HotpotQA training set and the first 100 conversations (452 wizard utterances) from the "test seen" set of WoW. Our set of experiments cost about 50 USD from OpenAI and about 50 USD from Together.ai.

A.2 Prompts

Tables 3 & 4 show the prompts we use for response generation on WoW and HotpotQA, respectively. While we experimented with Chain-of-Thought (Wei et al., 2022) for these prompts, they did not outperform zero-shot prompting for LLaMA 3.1 8B and Mistral-7B-Instruct, so we use zero-shot prompting throughout our experiments to keep the settings as simple as possible.

[*] https://platform.openai.com/docs/models/gpt-4o-mini

[*] https://docs.together.ai/docs/chat-models

A.3 Additional Plots

Figures 10, 11, and 12 extend Figure 3, with knowledge precision, knowledge recall, and knowledge F1 plotted against response/answer F1 for WoW and HotpotQA, respectively, for all three LLM generators. Figure 11 shows that knowledge recall correlates strongly with response/answer F1 for both GPT-4o-mini and LLaMA 3.1 8B on both WoW and HotpotQA, as well as for Mistral 7B-Instruct on WoW, while Figure 12 shows that knowledge F1 has the stronger correlation for Mistral 7B-Instruct on HotpotQA.

A.4 Noisiness of Wizard of Wikipedia vs. HotpotQA

To intuitively compare the noisiness of WoW vs. HotpotQA, for each example, we feed each *individual* candidate knowledge sentence, regardless of gold or distractor status, to the GPT-4o-mini generator model and measure the answer F1 score to measure how each individual knowledge sentence affects the generation performance. As Figure 7 shows, WoW knowledge sentences result in a wide, continuous distribution of response F1 scores, indicating that even the "distractor" knowledge positively contributes to the gold response to some extent. In contrast, we find that 70% of HotpotQA sentences are true distractors that cause the generator to produce incorrect answers with 0 F1 score, while 20% of sentences result in correct answers. These results indicate that WoW is a much noisier dataset than HotpotQA.

A.5 Noisy HotpotQA

As we discuss in Section 4.1, the WoW gold knowledge annotations are noisy in that only one sentence can be seelcted as gold knowledge, even if the other retrieved sentences are still relevant. To test the impact of such knowledge annotation noise on our RAG simulation results, we deliberately inject noise into HotpotQA by removing one sentence from the gold knowledge set for each example to create a *noisy HotpotQA* dataset. As a result, the average number of gold knowledge sentences per example drops from 2.4 to 1.4, and a small portion of the "distractor" knowledge in the noisy dataset is actually gold knowledge from the original HotpotQA (i.e. false-negative knowledge).

As Figures 5 & 6 show, the injected noise drastically impairs the knowledge selector's efficacy: even given a "perfect" knowledge selector with a

knowledge precision score of 100%, the answer F1 still underperforms the "full knowledge" setting. This is because our knowledge precision and recall scores are computed based on the labeled gold knowledge sentences, so if the labels are noisy, the knowledge metrics are corrupted, while the answer F1 is intact. Note that Figure 6 presents a non-monotonic trend in improving knowledge selector, just as we observed in Section 5 for WoW, verifying that noisy knowledge annotations cause this non-monotonic behavior.

We argue that the simulation results on WoW and noisy HotpotQA are more reflective of real-world RAG applications than the original (clean) HotpotQA. In a real application, there is no way to know the test-time knowledge recall and precision, and it is likely to be slightly different than the training-time knowledge recall and precision.

A.6 Length-Constrained Knowledge Selection and Answer Generation

To study the effect of length-constrained knowledge inputs on answer generation performance, we limit the number of knowledge sentences to k by randomly sub-sampling k sentences when more than k sentences are selected by the simulated knowledge selector*. We choose $k = 3$ for both WoW and HotpotQA because their average number of candidate knowledge sentences *before random sampling* are 7.1 and 9.5 respectively.

As Figure 8 shows, the length of the knowledge input mostly correlates with knowledge precision. As Figure 9 shows, constraining the knowledge input size pushes the points in the upper-left corner of the plots in Figure 2 into the rest of the space while keeping the overall trend of the points the same. Our observations in Section 5 hold even with length-constrained knowledge selection: a sufficiently high knowledge recall score is a prerequisite for a knowledge selector outperform the "full knowledge" setting.

*This random sampling does not break the experiment because our knowledge retriever and selector are simulated via random sampling, which is not the case for actual model-based retrievers and selectors.

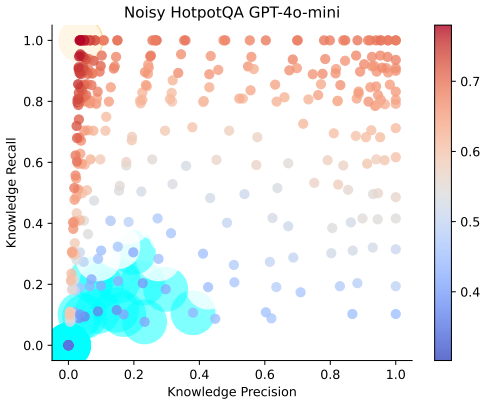

Figure 5: Scatter plot of answer F1 versus the knowledge precision for GPT-4o-mini on **noisy** HotpotQA.

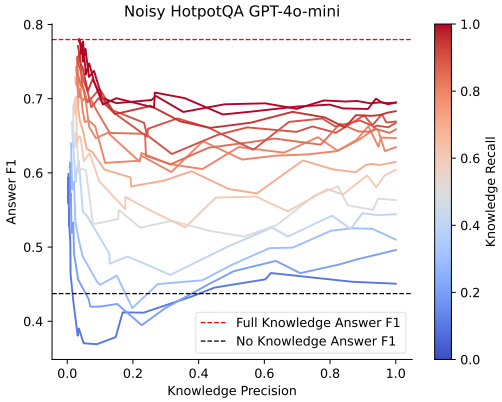

Figure 6: Color contours of answer F1 versus the knowledge precision for GPT-4o-mini on **noisy** HotpotQA. Each contour represents a different knowledge recall score; moving left to right visualizes improving the performance (precision) of the knowledge selector.

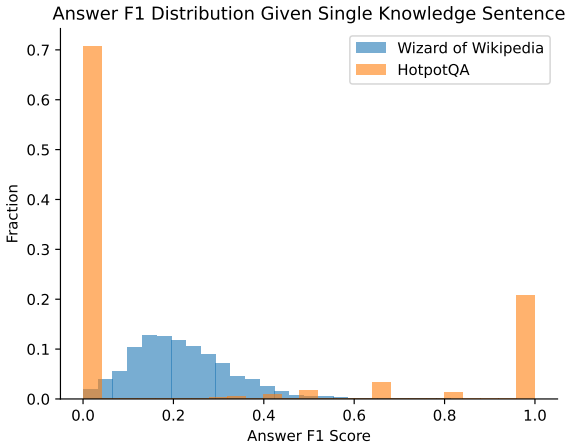

Figure 7: Histogram of answer F1 distributions of Wizard of Wikipedia and HotpotQA by feeding each *individual* candidate sentence to the GPT-4o-mini generator.

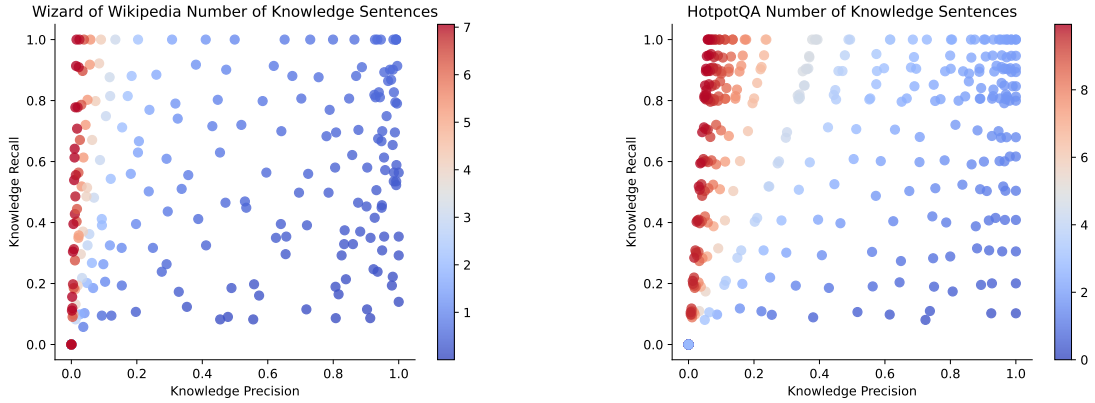

Figure 8: Scatter plot of the number of selected knowledge sentences versus the knowledge precision on Wizard of Wikipedia (left) and HotpotQA (right). *Each data point corresponds to a full experiment on the entire sampled dataset.*

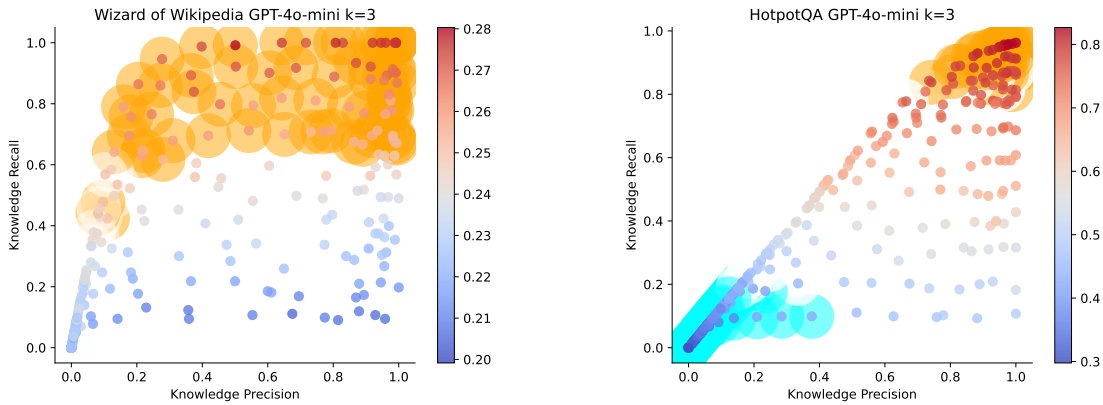

Figure 9: Scatter plot of answer F1 versus the knowledge precision for GPT-4o-mini on Wizard of Wikipedia (left) and HotpotQA (right) by limiting the number of knowledge sentences up to k=3. The dots highlighted in orange indicate settings outperforming the "full knowledge" setting (without constraining the number of sentences for fair comparison with other figures), while those highlighted in cyan indicate settings underperforming the "no knowledge" setting. *Each figure is a meta-experiment, and each data point corresponds to a full experiment on the entire sampled dataset.*

Prompt

```
The following is the conversation between the "Wizard",
a knowledgable speaker who can access to Wikipedia
knowledge sentences to chat to with the "Apprentice",
who does not have access to Wikipedia.

The conversation is about “{{persona}}”.
{% if history %}
Here is the conversation history:
{% for turn in history %}
{{turn.speaker}}: {{turn.text}}
{% endfor %}
{% endif %}

{% if context %}
Here are some retrieved Wikipedia knowledge for the
Wizard. The Wizard can choose any subset of the fol-
lowing knowledge. It’s also allowed to not choosing any
of them.

{% for evidence in context %}
Title: {{ evidence.title }}
Sentences:
{% for sentence in evidence.sentences %}
- {{ sentence }}
{% endfor %}

{% endfor %}
{% endif %}

Given the knowledge above, make a very brief, such as
one sentence, natural response for the Wizard.
Not all information in the chosen knowledge has to be
used in the response. Do not include the speaker’s name
in the response.
The Wizard’s response is:
```

Table 3: Jinja2 prompt template for Wizard of Wikipedia.

Prompt

```
Answer this question from HotpotQA with a response
that is as short as possible, e.g. one word:
{{ question }}
{% if context %}
Use the following support evidence to answer:
{% for evidence in context %}
Title: {{ evidence.title }}
Sentences:
{% for sentence in evidence.sentences %}
- sentence
{% endfor %}

{% endfor %}
{% endif %}
```

Table 4: Jinja2 prompt template for HotpotQA.

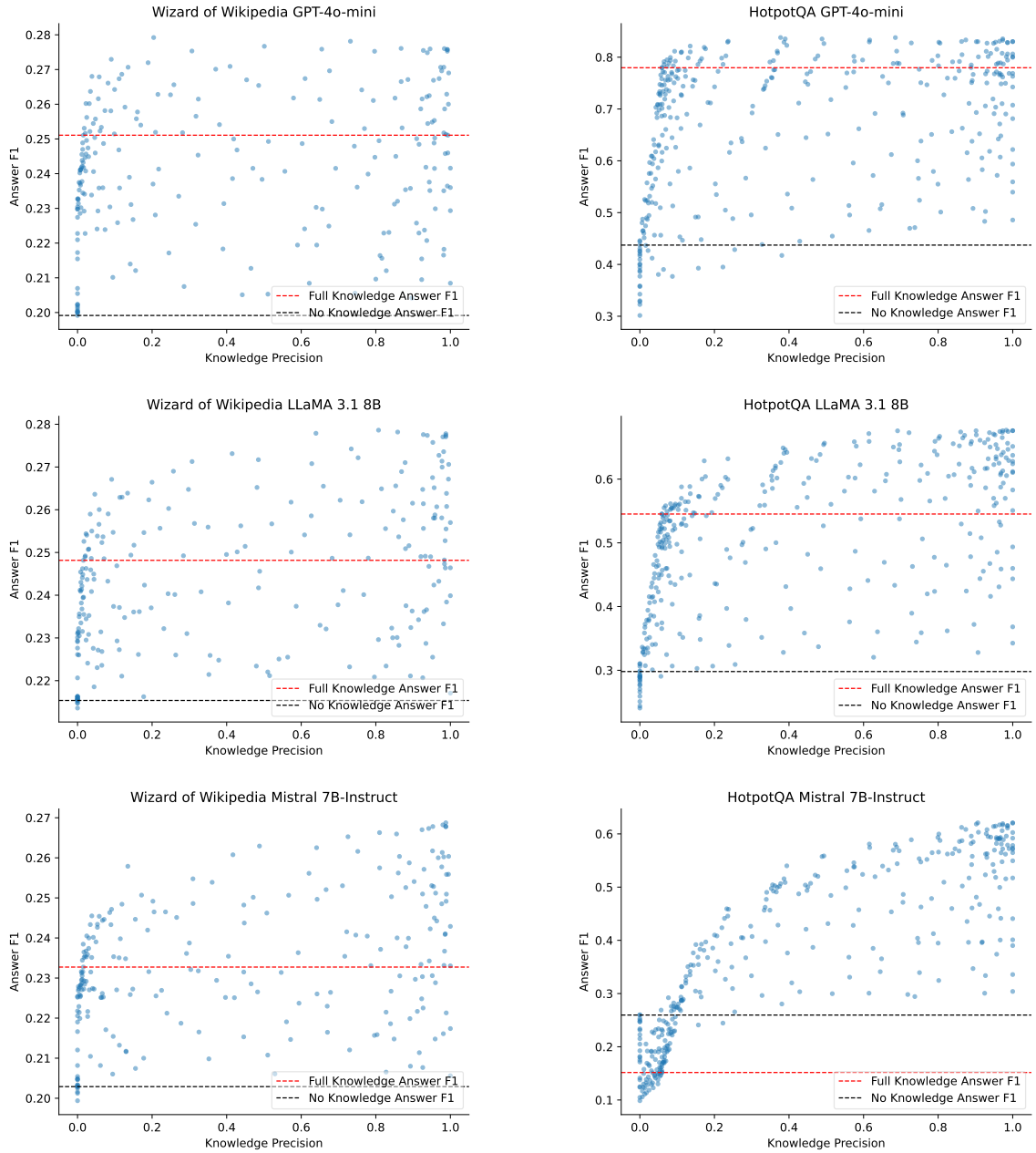

Figure 10: Scatter plot of answer F1 versus the knowledge precision for GPT-4o-mini (top), LLaMA 3.1 8B (middle), and Mistral 7B-Instruct (bottom); the left column shows results on WoW, and the right shows HotpotQA. *Each data point corresponds to a full experiment on the entire sampled dataset.*

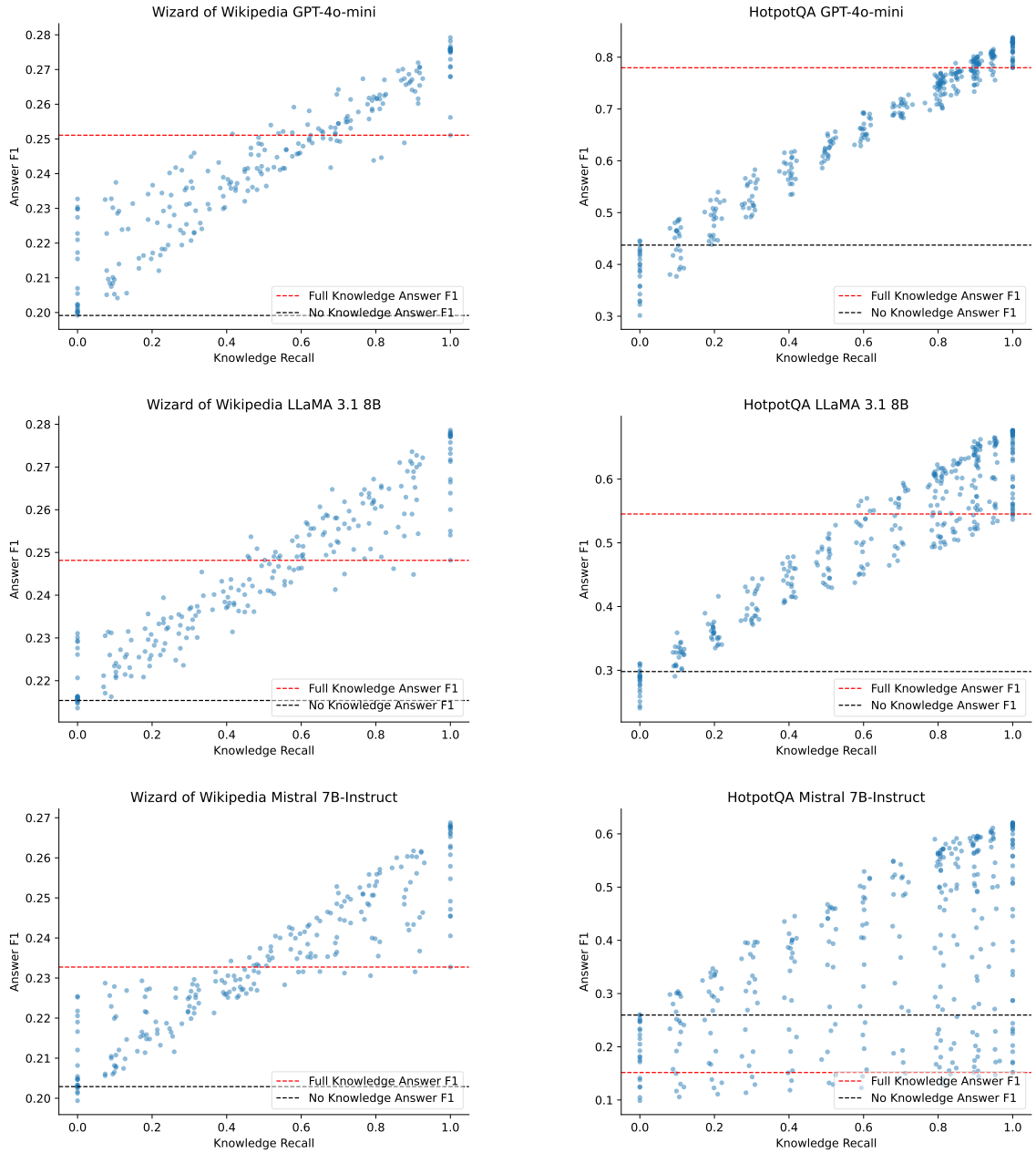

Figure 11: Scatter plot of answer F1 versus the knowledge recall for GPT-4o-mini (top), LLaMA 3.1 8B (middle), and Mistral 7B-Instruct (bottom); the left column shows results on WoW, and the right shows HotpotQA. *Each data point corresponds to a full experiment on the entire sampled dataset.*

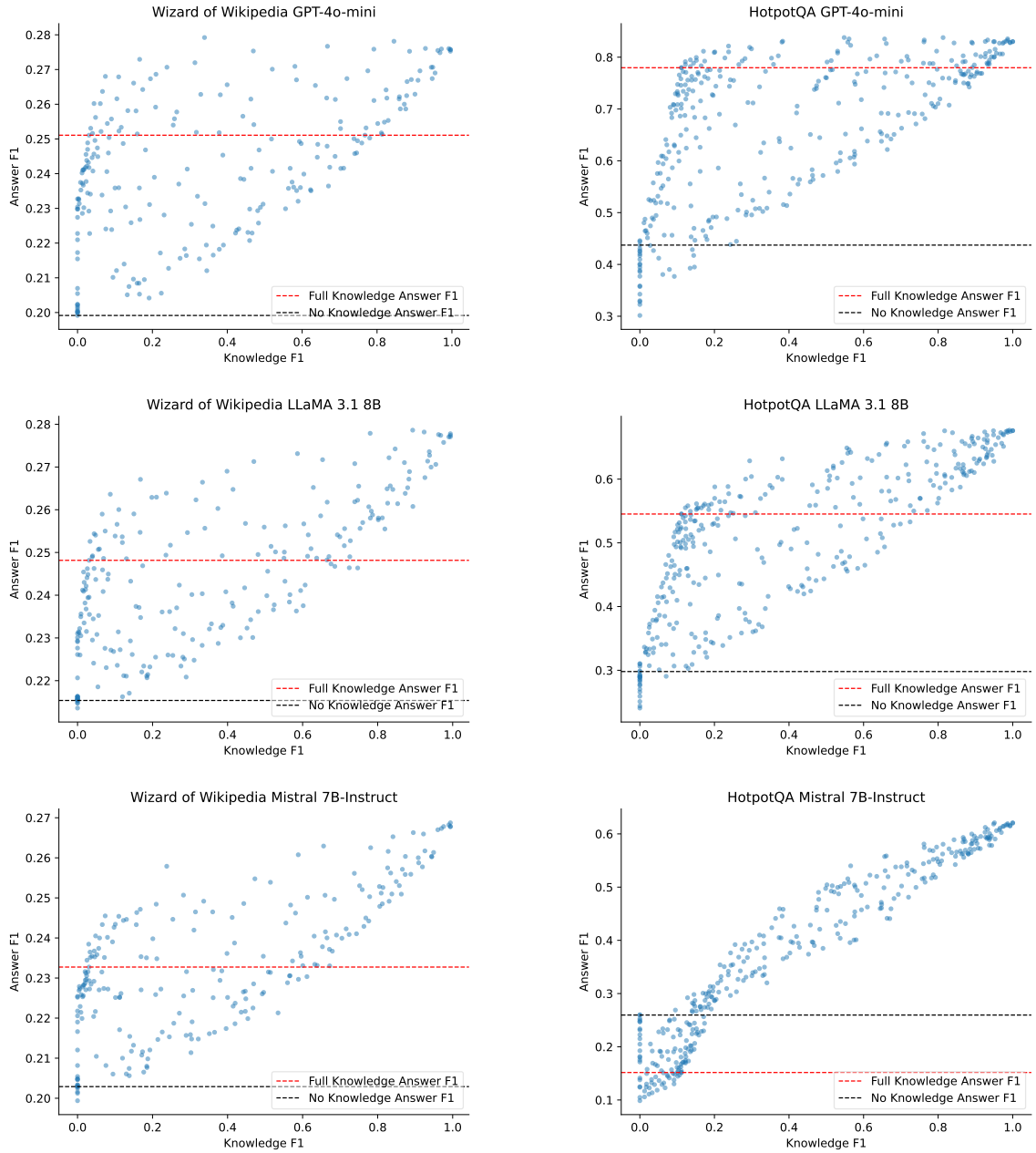

Figure 12: Scatter plot of answer F1 versus the knowledge F1 for GPT-4o-mini (top), LLaMA 3.1 8B (middle), and Mistral 7B-Instruct (bottom); the left column shows results on WoW, and the right shows HotpotQA. *Each data point corresponds to a full experiment on the entire sampled dataset.*