



Evaluating Retrieval Quality in Retrieval-Augmented Generation

Alireza Salemi

University of Massachusetts Amherst
Amherst, MA, United States
asalemi@cs.umass.edu

Hamed Zamani

University of Massachusetts Amherst
Amherst, MA, United States
zamani@cs.umass.edu

ABSTRACT

Evaluating retrieval-augmented generation (RAG) presents challenges, particularly for retrieval models within these systems. Traditional end-to-end evaluation methods are computationally expensive. Furthermore, evaluation of the retrieval model’s performance based on query-document relevance labels shows a small correlation with the RAG system’s downstream performance. We propose a novel evaluation approach, eRAG, where each document in the retrieval list is individually utilized by the large language model within the RAG system. The output generated for each document is then evaluated based on the downstream task ground truth labels. In this manner, the downstream performance for each document serves as its relevance label. We employ various downstream task metrics to obtain document-level annotations and aggregate them using set-based or ranking metrics. Extensive experiments on a wide range of datasets demonstrate that eRAG achieves a higher correlation with downstream RAG performance compared to baseline methods, with improvements in Kendall’s τ correlation ranging from 0.168 to 0.494. Additionally, eRAG offers significant computational advantages, improving runtime and consuming up to 50 times less GPU memory than end-to-end evaluation.

CCS CONCEPTS

- **Computing methodologies** → **Natural language generation**;
- **Information systems** → **Evaluation of retrieval results**.

KEYWORDS

Evaluation; retrieval quality; retrieval-augmented generation

ACM Reference Format:

Alireza Salemi and Hamed Zamani. 2024. Evaluating Retrieval Quality in Retrieval-Augmented Generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR ’24)*, July 14–18, 2024, Washington, DC, USA. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3626772.3657957>

1 INTRODUCTION

Retrieval-augmented generation (RAG) has emerged as a prominent approach in natural language processing, combining the strengths of retrieval and generation models [35], with use cases in decreasing hallucination [1, 29], knowledge-grounding [9, 16, 34], and personalization [25, 26]. Evaluating RAG systems is important as it ensures the effectiveness of integrating retrieval-based methods

with generative models [10, 23]. Traditionally, RAG evaluation has primarily relied on end-to-end assessment, which entails comparing the generated output with one or more ground truth references [20]. While this is crucial, it presents several limitations, especially, for evaluating retrieval models in RAG systems.

First, end-to-end evaluation lacks transparency regarding which retrieved document contributed to the generated output, hindering interpretability of the system’s behavior. Secondly, it is resource-intensive, consuming significant time and computational power, particularly when dealing with a large set of retrieval results consumed by the LLM. To process long input sequences resulting from the utilization of all retrieved documents by the LLM, GPUs with substantial memory capacities are essential for end-to-end evaluation. Moreover, many ranking systems rely on interleaving (i.e., replacing one or more documents in the result list) for evaluation and optimization, which further complicates the evaluation, as slight variations in retrieval results necessitate re-computation of the RAG pipeline. Finally, optimizing ranking models often requires document-level feedback, such as user clicks [3, 6]. However, end-to-end evaluation only provides list-level feedback for the retrieval results. That said, this paper studies retrieval evaluation in RAG.

Human annotations can be a potential solution for evaluating retrieval models in RAG, however, accurate annotations are often challenging and costly to obtain. More recently, with the emergence of large language models (LLMs) and their advanced capabilities in reasoning and text comprehension, they have been utilized to annotate documents for retrieval evaluation [10, 23]. Nevertheless, these approaches predominantly evaluate the retriever in RAG systems based on human preferences, whereas the primary objective of the retrieval model in RAG is to serve the LLM that leverages the retrieved results [35]. That said, our extensive investigation on a diverse set of RAG systems for open-domain question answering, fact verification, and dialogue systems reveals that employing human annotations, such as the *provenance* labels in the KILT benchmark [20], for evaluating the retrieval models within a RAG system exhibits only a minor correlation with the downstream RAG performance. This indicates a lack of meaningful relationship between the evaluated metrics and the downstream performance of RAG.

In this paper, we propose eRAG, a new approach for evaluating retrievers in RAG systems, where we apply the LLM in RAG system on each document in the retrieval result list individually and use the LLM’s output to provide document-level annotations. These annotations can be obtained using any arbitrary downstream task metric, such as accuracy, exact match, or ROUGE [17]. We can then apply a set-based or ranking metric as an aggregation function to obtain a single evaluation score for each retrieval result list.

We evaluate our proposed approach on question answering, fact-checking, and dialogue generation from the knowledge-intensive language tasks (KILT) benchmark [20]. Our results demonstrate that



This work is licensed under a Creative Commons Attribution International 4.0 License.

SIGIR ’24, July 14–18, 2024, Washington, DC, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0431-4/24/07.
<https://doi.org/10.1145/3626772.3657957>

our proposed approach achieves the highest correlation with the downstream performance of the RAG system in comparison with the baselines. Specifically, we observe an absolute improvement in Kendall’s tau correlation ranging between 0.168 and 0.494 across the evaluated datasets. Furthermore, we investigate the impact of different retrieval augmentation methods, the quantity of retrieved documents, and the LLM size on correlation. Finally, we demonstrate that our approach offers significant computational advantages, consuming up to 50 times less memory compared to end-to-end evaluation. To facilitate research in this domain, we make eRAG’s implementation publicly available at: <https://github.com/alirezasailemi7/eRAG>.

2 EVALUATING RETRIEVERS IN RAG

Generally, two predominant methods are used for obtaining relevance labels for retrieval evaluation. The first approach involves human judgment to assess the relevance of a query to documents within a corpus. The main issue with this approach is that human annotation can be costly and is often impractical for evaluating all documents in a corpus [28]. Moreover, human annotation relies on human preferences to judge the relevance of documents to a query. However, a document deemed relevant based on human preferences may not be useful for an LLM in fulfilling its task.

The second approach utilizes the downstream ground truth output associated with the query to provide weak relevance labels. In this method, a retrieved document containing the downstream ground truth is considered relevant [8, 14, 24, 27]. This method also presents its own challenges. This approach is impractical, particularly in scenarios where the task involves long-text generation or text classification, as downstream task labels might not exist within documents. Also, one document can be useful for an LLM in fulfilling its task without containing the ground truth labels.

Even though we are not aware any work that use LLMs for evaluating retrieval models in RAG, LLMs can be leveraged to label documents based on their relevance to a query. Inspired by Thomas et al. [30], the LLM functions as a binary classifier, indicating whether a document is relevant to the query or not. The mentioned challenges persist even with the judgment of LLMs, especially if the LLM responsible for labeling differs from the LLM in the RAG pipeline. Besides, employing LLMs as judges in this scenario can pose challenges due to the computational cost of running them on a large set of retrieved documents and memory constraints.

To mitigate these problems, we propose eRAG, a novel approach that involves utilizing the LLM in RAG system itself as the arbiter for generating labels to evaluate the retrieval model.

Using Downstream Large Language Model in RAG as Document Annotator. Consider a retrieval model \mathcal{R} that produces a ranked list \mathbf{R}_k with k documents for the LLM \mathcal{M} tasked with performing a specific task, utilizing a downstream evaluation function \mathcal{E}_M . The LLM \mathcal{M} takes a ranked list of documents as its input along with the query q , and generates an output represented as $\bar{y} = \mathcal{M}(q, \mathbf{R}_k)$. For the documents in \mathbf{R}_k , we feed each document individually to the LLM \mathcal{M} with the query and evaluate the generated answer to create the label for each document, expressed as:

$$\mathcal{G}_q[d] = \mathcal{E}_M(\mathcal{M}(q, \{d\}), y) \quad : \quad \forall d \in \mathbf{R}_k \quad (1)$$

where y is the expected downstream output for the query. We can employ the created \mathcal{G}_q to utilize any ranking metric to evaluate \mathcal{R} .

Note that the runtime cost of a vanilla transformer [32] scales quadratically with its input length. Consequently, for end-to-end evaluation, the cost of running a transformer on a ranked list with k documents, with an average length of d , to generate an output with length l is $O(lk^2d^2)$. Conversely, in our approach, as each document is individually fed to the LLM for k times, the cost is $O(lkd^2)$, proving to be more efficient than end-to-end evaluation.

Retrieval Evaluation Metrics. For a ranked list \mathbf{R}_k , comprising k retrieved documents generated by a retrieval model \mathcal{R} , an evaluation metric \mathcal{E}_R assigns a score $\mathcal{E}_R(\mathbf{R}_k, \mathcal{G}_q) \in [0, 1]$, by comparing the ranked list with the relevance scores \mathcal{G}_q , which is a function that maps each document to a scalar relevance score for the document with respect to the query q (i.e., $\mathcal{G}_q(d) = s_d$). Various definitions exist for the evaluation metric \mathcal{E}_R ; in this paper, we examine Precision (P), Recall (R), Mean Average Precision (MAP), Mean Reciprocal Rank (MRR) [2], Normalized Discounted Cumulative Gain (NDCG) [11], and Hit Rate. Note that when dealing with non-binary relevance labels, precision considers the average value of relevance labels, while Hit Ratio considers the maximum value among them.

3 EXPERIMENTS

3.1 Setup

Datasets and Evaluation. We use Natural Questions (NQ) [15], TriviaQA [13], HotpotQA [33], FEVER [31], and Wizard of Wikipedia (WoW) [4] datasets from the KILT [20] benchmark. Due to the unavailability of ground truth labels for the test set, we utilize the publicly accessible validation set. As the retrieval corpus, we employ the Wikipedia dump of the KILT benchmark and adhere to the preprocessing outlined by Karpukhin et al. [14], where each document is segmented into passages, each constrained to a maximum length of 100 words. The concatenation of the article title and passage is used as a document. The KILT benchmark furnishes document-level relevance labels (called Provenance) for its datasets, and these are employed for evaluating retrieval performance. In line with our preprocessing method, we define all passages within a positive document as positive passages for our evaluation. For relevance evaluation using an LLM, we employ Mistral¹ [12] to annotate each document within the retrieved list, determining whether it is relevant to the query or not. We adopt the metrics recommended by the KILT benchmark, namely Exact Match (EM) for NQ, TriviaQA, and HotpotQA, Accuracy for FEVER, and F1 for the WoW dataset.

Experiments Configuration. In all experiments, unless explicitly stated otherwise, we employ T5-small [21] with Fusion-in-Decoder (FiD) [9] as the LLM. We employ AdamW [19] with a weight decay of 10^{-2} and a learning rate of 5×10^{-5} for 10 epochs, incorporating linear warmup for the initial 5% of training steps. The effective batch size is set to 64. Each model is trained using an A100 Nvidia GPU. For document retrieval during training, we utilize BM25 [22] implemented in Pyserini [18] to retrieve 50 documents to augment the input with them. For fast vector search in dense retrieval with Contriever² [7], we use Faiss [5] flat index.

¹<https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2>

²<https://huggingface.co/facebook/contriever>

Table 1: The correlation between each evaluation approach and the downstream performance of the LLM. T5-small with FiD with 50 retrieved documents is used. We do not report correlation for the Answers method for FEVER and WoW datasets because the answers to queries do not exist in the document since FEVER is a classification dataset and WoW is long-text generation. For the WoW dataset, we only report correlation on Precision and Hit Ratio because other metrics do not support non-integer relevance labels. Tau is Kendall’s tau and rho is Spearman’s rho.

Relevance Annotation	Metric	BM25										Contriever									
		NQ		TriviaQA		HotpotQA		FEVER		WoW		NQ		TriviaQA		HotpotQA		FEVER		WoW	
		tau	rho	tau	rho	tau	rho	tau	rho	tau	rho	tau	rho	tau	rho	tau	rho	tau	rho	tau	rho
Containing the Answer	MAP	0.349	0.417	0.298	0.364	0.359	0.423	-	-	-	-	0.303	0.366	0.265	0.325	0.379	0.429	-	-	-	-
	MRR	0.361	0.417	0.313	0.340	0.398	0.449	-	-	-	-	0.301	0.353	0.257	0.292	0.384	0.430	-	-	-	-
	NDCG	0.357	0.427	0.298	0.365	0.370	0.435	-	-	-	-	0.313	0.378	0.270	0.331	0.385	0.437	-	-	-	-
	P	0.353	0.411	0.276	0.333	0.396	0.454	-	-	-	-	0.346	0.403	0.283	0.340	0.406	0.449	-	-	-	-
	R	0.325	0.325	0.232	0.232	0.375	0.375	-	-	-	-	0.319	0.319	0.215	0.215	0.401	0.401	-	-	-	-
	Hit Ratio	0.325	0.325	0.232	0.232	0.375	0.375	-	-	-	-	0.319	0.319	0.215	0.215	0.401	0.401	-	-	-	-
KILT Provenance	MAP	0.181	0.218	0.142	0.172	0.007	0.009	0.026	0.032	0.015	0.021	0.161	0.196	0.113	0.137	0.128	0.155	0.045	0.056	0.055	0.080
	MRR	0.177	0.205	0.151	0.175	0.074	0.080	0.036	0.040	0.013	0.017	0.152	0.173	0.120	0.136	0.151	0.169	0.045	0.049	0.059	0.081
	NDCG	0.179	0.216	0.142	0.172	0.021	0.026	0.029	0.036	0.013	0.019	0.159	0.193	0.115	0.140	0.134	0.162	0.045	0.056	0.056	0.081
	P	0.163	0.192	0.140	0.165	0.139	0.164	0.043	0.051	0.011	0.015	0.131	0.157	0.108	0.130	0.181	0.215	0.033	0.040	0.045	0.064
	R	0.216	0.216	0.187	0.187	0.113	0.113	0.050	0.050	0.019	0.023	0.157	0.157	0.135	0.135	0.163	0.163	0.038	0.038	0.056	0.068
	Hit Ratio	0.216	0.216	0.187	0.187	0.113	0.113	0.050	0.050	0.019	0.023	0.157	0.157	0.135	0.135	0.163	0.163	0.038	0.038	0.056	0.068
Relevance Annotation with LLM (Mistral 7B)	MAP	0.045	0.055	0.176	0.216	0.034	0.042	0.018	0.022	-0.005	-0.008	0.032	0.039	0.174	0.213	0.051	0.063	0.021	0.026	-0.002	-0.003
	MRR	0.060	0.062	0.189	0.196	0.001	0.001	-0.021	-0.022	-0.008	-0.011	0.048	0.050	0.143	0.151	0.034	0.038	-0.007	-0.007	0.004	0.005
	NDCG	0.049	0.060	0.178	0.218	0.032	0.039	0.018	0.022	-0.006	-0.009	0.036	0.044	0.175	0.214	0.049	0.060	0.022	0.028	0.000	0.000
	P	0.028	0.034	0.137	0.166	-0.004	-0.006	0.021	0.025	-0.005	-0.008	0.002	0.003	0.138	0.167	0.010	0.013	0.014	0.017	-0.006	-0.010
	R	0.014	0.014	0.032	0.032	-0.016	-0.016	0.019	0.019	0.003	0.003	0.000	0.000	0.039	0.039	-0.042	-0.042	-0.017	-0.017	0.017	0.021
	Hit Ratio	0.014	0.014	0.032	0.032	-0.016	-0.016	0.019	0.019	0.003	0.003	0.000	0.000	0.039	0.039	-0.042	-0.042	-0.017	-0.017	0.017	0.021
eRAG Annotations	MAP	0.492	0.575	0.474	0.578	0.610	0.694	0.386	0.463	-	-	0.467	0.544	0.427	0.519	0.634	0.705	0.399	0.479	-	-
	MRR	0.503	0.577	0.486	0.553	0.629	0.695	0.592	0.611	-	-	0.466	0.537	0.424	0.495	0.639	0.698	0.481	0.504	-	-
	NDCG	0.505	0.590	0.486	0.592	0.612	0.697	0.404	0.484	-	-	0.481	0.560	0.440	0.536	0.635	0.705	0.403	0.484	-	-
	P ^a	0.529	0.598	0.484	0.577	0.594	0.663	0.329	0.391	0.504	0.669	0.522	0.586	0.482	0.571	0.633	0.695	0.378	0.449	0.540	0.712
	R	0.519	0.519	0.426	0.426	0.619	0.619	0.301	0.301	-	-	0.488	0.488	0.408	0.408	0.631	0.631	0.299	0.299	-	-
	Hit Ratio ^b	0.519	0.519	0.426	0.426	0.619	0.619	0.301	0.301	0.390	0.532	0.488	0.488	0.408	0.408	0.631	0.631	0.299	0.299	0.414	0.561

^a For non-integer relevance labels, precision is equal to average of the relevance labels.

^b For non-integer relevance labels, hit ratio is equal to maximum of the relevance labels.

3.2 Main Findings

How do different retrieval evaluation methods correlate with the end-to-end downstream performance in RAG? To compare the different evaluation strategies for evaluating retriever in RAG, we report the correlation between the scores generated for each method and the downstream performance of the LLM (i.e., T5-small with FiD and 50 retrieved documents) in Table 1. The results indicate that eRAG attains the highest correlation compared to other evaluation approaches. Furthermore, the results show that regardless of the retrieval model employed, eRAG consistently outperforms others in terms of correlation with the LLM’s downstream performance. Interestingly, the most common approaches, KILT Provenance and Annotation with LLMs, that are, document-level relevance labels and using LLMs to assign a relevance label to each retrieved document, have the lowest correlation with the downstream performance of the LLM. This finding confirms that the LLM as the consumer of the retrieved results in RAG is the best judge for the performance of the retrieval model.

How do different retrieval evaluation methods in RAG perform as the size of retrieval results increases? To address this, we varied the number of retrieved documents and computed the correlation between the metric with highest correlation for each method in Table 1 at each specified number of retrieved documents and the downstream performance of the LLM given that number of retrieved documents. For the sake of space, we limit our experiments to three datasets: NQ for question answering, FEVER for fact-checking, and WoW for long-text generation. The results of this experiment are shown in Figure 1. The outcomes of this experiment reveal that irrespective of the quantity of retrieved documents,

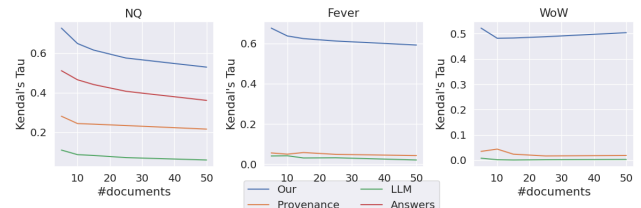


Figure 1: The correlation between evaluation approaches and the LLM’s downstream performance varying number of retrieved documents by BM25. T5-small with FiD is used. The metric with the highest correlation in Table 1 is used.

our suggested evaluation strategy consistently exhibits a higher correlation with the downstream performance of the LLM. Furthermore, the results illustrate that augmenting the number of retrieved documents leads to a decline in correlation—a intuitive observation, as all metrics assess each document-relevance label independently for scoring a ranked list, while the LLM uses information from the entirety of these documents to accomplish its task.

How does our method correlate with the downstream RAG performance as the size of large language models increases?

In addressing this question, we computed the correlation between our retrieval evaluation strategy and the downstream performance of the LMs with two distinct sizes (i.e., T5-small with FiD consisting of 60M and T5-base with FiD consisting of 220M parameters). For the sake of space, we limit our experiments to three datasets: NQ for question answering, FEVER for fact-checking, and WoW for long-text generation. The results illustrated in Figure 2 indicate that, for certain datasets, there is a higher correlation with the smaller LLM, while for others, a higher correlation is observed with the

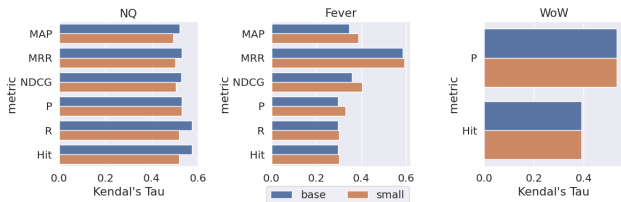


Figure 2: The correlation between eRAG and the downstream performance of different LLM sizes. In this experiment, T5-small (60M parameters) and T5-base (220M parameters) with FiD are used. The documents are retrieved using BM25.

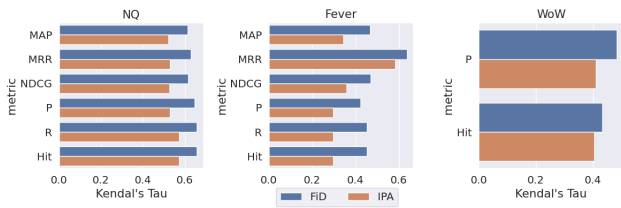


Figure 3: The correlation between eRAG and the downstream performance of FiD and IPA LLMs. T5-small with 10 documents retrieved by BM25 is used. The number of documents is chosen based on the limitations of the input size in IPA.

larger model. Nonetheless, in none of the cases is there a significant difference between the correlations, suggesting that the proposed approach is effective regardless of the LLM size.

How does different retrieval-augmentation approaches affect the correlation between eRAG and the downstream RAG performance? We applied eRAG to two LLMs. One LLM utilizes In-Prompt Augmentation (IPA), where the retrieved results are appended to the input of the LLM. The other LLM employs Fusion-in-Decoder (FiD) [9], wherein each retrieved document is individually processed by the encoder, and subsequently, the representations for all documents are concatenated together and fed to the decoder. For the sake of space, we limit our experiments to NQ for question answering, FEVER for fact-checking, and WoW for long-text generation. The correlation between eRAG and the outputs of each LLM is illustrated in Figure 3. Interestingly, the results suggest that although there is no significant difference between the correlation of eRAG with IPA and FiD LLMs, eRAG consistently exhibits a higher correlation with the FiD. This observation can be elucidated by considering the distinction between IPA and FiD methodologies. In IPA, all documents are concatenated together and then presented as a single input to the LLM. In contrast, FiD processes each document individually by feeding them separately to the LLM’s encoder. Given that our approach aligns more closely with FiD, we believe this alignment is a contributing factor to the higher correlation between eRAG and the downstream performance of FiD.

How much more efficient is eRAG compared to the end-to-end evaluation? Here, we consider two factors: inference time and memory consumption. For inference time, we compare the total time required for end-to-end evaluation to generate scores with the total time used by eRAG. In this experiment, we opt for the batch size of each approach to be as large as possible, maximizing the

Table 2: The runtime and memory consumption of eRAG in comparison with end-to-end evaluation. T5-small with FiD, consuming 50 documents is used.

Dataset	Runtime (GPU)		Memory Consumption (GPU)		
	E2E	eRAG	E2E	eRAG-Query	eRAG-Document
NQ	918 sec	351 sec	75.0 GB	4.9 GB	1.5 GB
TriviaQA	1819 sec	686 sec	46.2 GB	5.4 GB	1.5 GB
HotpotQA	1844 sec	712 sec	52.4 GB	5.5 GB	1.5 GB
FEVER	3395 sec	1044 sec	66.5 GB	4.1 GB	1.5 GB
WoW	912 sec	740 sec	47.9 GB	6.5 GB	1.5 GB

utilization of the entire GPU memory. The results of this experiment are reported in Table 2. The findings indicate that, on average, eRAG is 2.468 times faster than end-to-end evaluation. Further elaborating, the speedup for eRAG ranges from 1.232 to 3.252 times compared to end-to-end evaluation across the datasets, where the least speedup is for the long-text generation task (i.e., WoW).

To compare memory consumption between eRAG and end-to-end evaluation, we conducted two experiments. First, we compared the maximum memory required by end-to-end evaluation to assess a query with the maximum memory demanded by eRAG for the same evaluation. To carry out this comparison, we configured the batch size for end-to-end evaluation to 1, while for eRAG, we set it to the same number of documents used for one query by end-to-end evaluation (we call this query-level configuration). In the subsequent experiments, we set both batch sizes to 1 to assess the extent to which eRAG demonstrates superior memory efficiency compared to end-to-end evaluation under the most efficient configuration (we call this document-level configuration). The results of these experiments are reported in Table 2. The findings indicate that in the query-level configuration, eRAG exhibits between 7 to 15 times greater memory efficiency compared to end-to-end evaluation. Furthermore, in the document-level configuration, this efficiency gap widens, with eRAG demonstrating 30 to 48 times more memory efficiency than end-to-end evaluation across different dataset. In summary, these experiments suggest that eRAG is more efficient than end-to-end evaluation of a vanilla transformer, excelling in both inference time and memory utilization.

4 CONCLUSION

This paper explores various approaches for evaluating retrieval models within a RAG pipeline. Additionally, it introduces eRAG, a novel approach for evaluating retrieval models in the RAG pipeline. eRAG leverages the per-document performance of the LLM on the downstream task to generate relevance labels. The findings suggest that the proposed approach exhibits significantly higher correlation with the downstream performance of the LLM. Furthermore, eRAG demonstrates greater efficiency than end-to-end evaluation in terms of both memory consumption and inference time.

ACKNOWLEDGMENT

This work was supported in part by the Center for Intelligent Information Retrieval, in part by Lowe’s, and in part by an Amazon Research Award, Fall 2022 CFP. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

- [1] Garima Agrawal, Tharindu Kumarage, Zeyad Alghami, and Huan Liu. 2023. Can Knowledge Graphs Reduce Hallucinations in LLMs? : A Survey. arXiv:2311.07914 [cs.CL]
- [2] Nick Craswell. 2009. *Mean Reciprocal Rank*. Springer US, Boston, MA, 1703–1703. https://doi.org/10.1007/978-0-387-39940-9_488
- [3] Romain Deffayet, Philipp Hager, Jean-Michel Renders, and Maarten de Rijke. 2023. An Offline Metric for the Debaisedness of Click Models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (, Taipei, Taiwan.) (SIGIR '23). Association for Computing Machinery, New York, NY, USA, 558–568. <https://doi.org/10.1145/3539618.3591639>
- [4] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-Powered Conversational Agents. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=r1l73iRqKm>
- [5] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvassy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). arXiv:2401.08281 [cs.LG]
- [6] Fan Guo, Chao Liu, and Yi Min Wang. 2009. Efficient multiple-click models in web search. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining* (Barcelona, Spain) (WSDM '09). Association for Computing Machinery, New York, NY, USA, 124–131. <https://doi.org/10.1145/1498759.1498818>
- [7] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. *Transactions on Machine Learning Research* (2022). <https://openreview.net/forum?id=jKN1pXi7b0>
- [8] Gautier Izacard and Edouard Grave. 2021. Distilling Knowledge from Reader to Retriever for Question Answering. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=NTEz-6wysdb>
- [9] Gautier Izacard and Edouard Grave. 2021. Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, Paola Merlo, Jorg Tiedemann, and Reut Tsarfay (Eds.). Association for Computational Linguistics, Online, 874–880. <https://doi.org/10.18653/v1/2021.eacl-main.74>
- [10] Jithin James and Shahul Es. 2023. Ragas: Evaluation framework for your retrieval augmented generation (rag) pipelines. <https://github.com/explodinggradients/ragas>.
- [11] Kalervo Järvelin and Jaana Kekäläinen. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Athens, Greece) (SIGIR '00). Association for Computing Machinery, New York, NY, USA, 41–48. <https://doi.org/10.1145/345508.345545>
- [12] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. arXiv:2310.06825 [cs.CL]
- [13] 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)*, Regina Barzilay and Min-Yen Kan (Eds.). Association for Computational Linguistics, Vancouver, Canada, 1601–1611. <https://doi.org/10.18653/v1/P17-1147>
- [14] 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)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 6769–6781. <https://doi.org/10.18653/v1/2020.emnlp-main.550>
- [15] 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 (2019), 452–466. https://doi.org/10.1162/tacl_a_00276
- [16] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) (NIPS'20). Curran Associates Inc., Red Hook, NY, USA, Article 793, 16 pages.
- [17] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*. Association for Computational Linguistics, Barcelona, Spain, 74–81. <https://aclanthology.org/W04-1013>
- [18] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) (SIGIR '21). Association for Computing Machinery, New York, NY, USA, 2356–2362. <https://doi.org/10.1145/3404835.3463238>
- [19] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=Bkg6RiCqY7>
- [20] Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Mailard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a Benchmark for Knowledge Intensive Language Tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (Eds.). Association for Computational Linguistics, Online, 2523–2544. <https://doi.org/10.18653/v1/2021.naacl-main.200>
- [21] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* 21, 1, Article 140 (jan 2020), 67 pages.
- [22] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Text Retrieval Conference*. <https://api.semanticscholar.org/CorpusID:3946054>
- [23] Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2023. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. arXiv:2311.09476 [cs.CL]
- [24] Alireza Salemi, Juan Altmayer Pizzorno, and Hamed Zamani. 2023. A Symmetric Dual Encoding Dense Retrieval Framework for Knowledge-Intensive Visual Question Answering. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) (SIGIR '23). Association for Computing Machinery, New York, NY, USA, 110–120. <https://doi.org/10.1145/3539618.3591629>
- [25] Alireza Salemi, Surya Kallumadi, and Hamed Zamani. 2024. Optimization Methods for Personalizing Large Language Models through Retrieval Augmentation. In *Proceedings of the 47th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington, DC, USA) (SIGIR '24). (to appear).
- [26] Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2023. LaMP: When Large Language Models Meet Personalization. arXiv:2304.11406 [cs.CL]
- [27] Alireza Salemi, Mahta Rafiee, and Hamed Zamani. 2023. Pre-Training Multi-Modal Dense Retrievers for Outside-Knowledge Visual Question Answering. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval* (Taipei, Taiwan) (ICTIR '23). Association for Computing Machinery, New York, NY, USA, 169–176. <https://doi.org/10.1145/3578337.3605137>
- [28] Donia Scott, Rossano Barone, and Rob Koeling. 2012. Corpus Annotation as a Scientific Task. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (Eds.). European Language Resources Association (ELRA), Istanbul, Turkey, 1481–1485. http://www.lrec-conf.org/proceedings/lrec2012/pdf/569_Paper.pdf
- [29] 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*. Association for Computational Linguistics, Punta Cana, Dominican Republic, 3784–3803. <https://doi.org/10.18653/v1/2021.findings-emnlp.320>
- [30] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2023. Large language models can accurately predict searcher preferences. arXiv:2309.10621 [cs.IR]
- [31] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a Large-scale Dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Marilyn Walker, Heng Ji, and Amanda Stent (Eds.). Association for Computational Linguistics, New Orleans, Louisiana, 809–819. <https://doi.org/10.18653/v1/N18-1074>
- [32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
- [33] 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*, Ellen Riloff,

- David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, Brussels, Belgium, 2369–2380. <https://doi.org/10.18653/v1/D18-1259>
- [34] Hamed Zamani and Michael Bendersky. 2024. Stochastic RAG: End-to-End Retrieval-Augmented Generation through Expected Utility Maximization. In *Proceedings of the 47th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24)*. (to appear).
- [35] Hamed Zamani, Fernando Diaz, Mostafa Dehghani, Donald Metzler, and Michael Bendersky. 2022. Retrieval-Enhanced Machine Learning. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (Madrid, Spain) (SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2875–2886. <https://doi.org/10.1145/3477495.3531722>