2024 10th International Conference on Web Research (ICWR)

Hybrid Retrieval-Augmented Generation Approach for LLMs Query Response Enhancement

Pouria Omrani∗‡, Alireza Hosseini†‡, Kiana Hooshanfar†‡, Zahra Ebrahimian†‡, Ramin Toosi†‡, Mohammad Ali Akhaee†
‡Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran
†School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran
‡Adak Vira Iranian Rahjoo Company, Tehran, Iran
{pouria.omrani@ieee.org, arhosseini77@ut.ac.ir, k.hooshanfar@ut.ac.ir, z.ebrahimian@ut.ac.ir, r.toosi@ut.ac.ir, akhaee@ut.ac.ir}

2024 10th International Conference on Web Research (ICWR) | 979-8-3503-9498-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICWR61162.2024.10533345

***Abstract*— In the domain of Natural Language Pro- cessing (NLP), the integration of Large Language Models (LLMs) with Retrieval-Augmented Generation (RAG) represents a significant advancement towards enhancing the depth and relevance of model-generated responses. This paper introduces a novel hybrid RAG framework that synergizes the Sentence-Window and Parent-Child methodologies with an innovative re-ranking mechanism, aimed at optimizing the query response capabilities of LLMs. By leveraging exter- nal knowledge sources more effectively, the proposed method enriches LLM outputs with greater accuracy, relevance, and information fidelity. We subject our hybrid model to rigorous evaluation against bench- mark datasets and metrics, demonstrating its superior performance over existing state-of-the-art RAG tech- niques. The results highlight our method's enhanced ability to generate responses that are not only contex- tually appropriate but also demonstrate a high degree of faithfulness to the source material, thereby setting a new standard for query response enhancement in LLMs. Our study underscores the potential of hybrid RAG models in refining the interaction between LLMs and external knowledge, paving the way for future research in the field of NLP.**

Keywords*— LLM, Generative AI, NLP, Retrieval Augmented Generation (RAG).

I. Introduction

NLP is a domain within artificial intelligence (AI) that focuses on the interaction between computers and humans using natural language. The ultimate objective of NLP is to enable computers to understand, interpret and generate human languages in a way that is both valuable and meaningful [1]. Applications of NLP are widespread and diverse, ranging from speech recognition systems [2] to sentiment analysis [3] and chatbots [4].

LLMs represent a significant advancement in the NLP field. LLMs, such as GPT [5] and LLama [6] models, are trained on vast datasets of text, allowing them to generate coherent, contextually relevant text based on the input they receive. These models have significantly pushed the boundaries of what's possible in NLP, demonstrating impressive capabilities in text generation, conversation, summarization, and even generating programming code [7]. Given the emergence of LLMs as a prevailing trend in the field of NLP, many studies have been conducted in this domain. Li et al.[8] introduced a novel LLM- executable clinical guidance tree structure and proposed a new medical decision-making dataset with a decision- retrieval-based generation framework. Zeng et al.[9] pro- vided a comprehensive overview of the wide range of ap- plications for LLMs within social networks, discussing all potential challenges one might encounter. Trad et al.[10] presented a comparative analysis of prompt engineering and fine-tuning techniques for LLMs, explored prompt- engineering strategies for phishing URL detection, and investigated the fine-tuning of text-generation LLMs in this domain.

Despite the impressive achievements of LLMs, they face several significant challenges that limit their effectiveness and applicability in real-world scenarios. To address the limitations related to resource use in training LLMs, Hu et al.[11] proposed a training technique that significantly reduces the number of trainable parameters.

Furthermore, although LLMs are proficient at generat- ing coherent text, they sometimes struggle with under- standing the context or intent behind specific queries or tasks, leading to responses that may be irrelevant or off- topic. Additionally, because LLMs are trained on static datasets, their knowledge is effectively frozen at the point of training, preventing them from accessing or incorpo- rating up-to-date information. This limitation restricts their usefulness for tasks requiring current knowledge [12]. To address these issues, the RAG approach has emerged as a promising solution. RAG combines the generative capabilities of LLMs with real-time information retrieval from external databases [13]. This technique allows LLMs to access and integrate up-to-date information or domain- specific knowledge that was not available during their initial training phase. By doing so, RAG models can generate responses that are not only contextually rele- vant but also reflect current events, trends and facts, significantly enhancing their utility for a wide range of applications, from automated news reporting to person- alized education and advice. Finardi et al.[14] explored two retrieval techniques (naive and sentence window) to optimize the integration of retrieved information into the LLM generation process. Cheng et al.[15] proposed a novel framework that leverages iterative RAG to create an unbounded memory pool from its output for improved text generation. In another work, authors introduced enhancements to the RAG model to boost its adapt- ability in specialized domains for open-domain question answering [16]. Chen et al.[17] introduced the Multimodal RAG, a novel approach that enhanced language generation by incorporating a non-parametric multimodal memory, enabling the model to retrieve and utilize knowledge from both text and images.

Leveraging the success of advanced RAG techniques, such as the sentence-window and parent-child methods, this paper introduces a novel hybrid approach that in- tegrates both techniques alongside re-ranking modules. These modules are designed to evaluate and reorder the retrieved results. Furthermore, the paper provides a com- parative analysis using benchmark metrics, demonstrating that the proposed

method outperforms previous state-of- the-art approaches. Thus, the main contributions of this paper are highlighted as follows:

- Proposing a hybrid RAG that incorporates both sentence window and parent-child RAGs.
- Providing a comparative analysis demonstrating that the proposed method outperforms previous state-of-the-art RAG methods.

The remainder of this paper is organized as follows: Section II begins by describing naive RAG and advanced techniques, then proposes a hybrid approach for infor- mation retrieval. Subsequently, Section III describes the benchmark dataset and metrics, evaluates the proposed method and compares it to previous state-of-the-art meth- ods. Finally, Section IV concludes the paper and provides suggestions for future research.

II. Methods

This section, first introduces Naive RAG and advanced RAG techniques (Sentence-Window and Parent-Child). After that, proposed method is described as a hybrid approach, which is designed for enhancing query responses by integrating contextual data retrieval with LLMs.

A. Naive RAG

The Naive RAG pipeline, which is shown in Fig. 1, begins with a query input, which is processed by an embedding model to transform the query into a vector representation. This representation is then used to search a vector store index, which is a structured collection of vectors representing data from a database. The database is queried based on the closest matching vectors to the query vector. The retrieved data provides context that is fed into a LLM, which uses the context to generate an answer relevant to the original query [13].

B. Sentence-Window

Sentence-Window RAG, as seen in Fig.2, focuses on smaller units of text within a larger context for in- formation retrieval. In this process, documents are first broken down into chunks, which are then further distilled into embeddings of even smaller pieces, like sentences or sentence fragments, for finer retrieval granularity. These embeddings are vector representations designed for re- trieval tasks. When a query is made, the system identifies the top k relevant embeddings (the smaller chunks) and maps them back to their corresponding larger chunks to maintain context. These context-rich chunks are then fed into a LLM, which uses the detailed, specific information from the embeddings along with the broader context from the chunks to generate a coherent and contextually in- formed answer. This approach allows for precise retrieval of information with enough context to generate accurate and relevant responses [14].

C. Parent-Child

In Parent-Child RAG, as illustrated in Fig.3, a child chunk refers to a segment of information extracted from a larger dataset that specifically relates to the user's query. On the other hand, a parent chunk denotes a broader section of the dataset encompassing one or more related child chunks, thereby offering additional context. The pipeline operates as follows: Upon receiving a query, the system searches a vectorized store of information to find and rank child chunks by relevance. The top child chunks are then matched with their respective parent chunks, which are used by a LLM as a contextual foundation to understand and integrate the detailed information from the child chunks. This integrated understanding enables the LLM to generate a comprehensive and contextually informed answer [18].

D. Proposed Method

In the proposed pipeline, which is shown in Fig.4, integrating sentence-window and parent-child RAG models, the process begins with an input query Q that activates two simultaneous retrieval mechanisms. The sentence- window approach targets accurate information retrieval by selecting the top-K matches from embeddings derived from smaller text units within the documents D, ensuring a granular focus on the most relevant snippets of text directly related to the query. At the same time, the parent-child retrieval method works on a broader level, pinpointing the top-K matches from child chunks and their respective parent chunks, which provide additional context.

According to (1), the sentence-window approach employs a scoring function f_{sw}. The embeddings, $e_{sw,i}$, rep- resent smaller units of text derived from the documents, where i indexes these embeddings.

$$SW\ (Q, D, K) = \ top - K\ f_{sw}\big(Q, e_{sw,i}\big),\ e_{sw,i} \in E \tag{1}$$

The parent-child retrieval mechanism, as seen in (2), utilizes a scoring function f_{pc}. The retrieval process matches child chunks, c_j, with their corresponding parent chunks, p_j, where j indexes these pairs to ensure the context is maintained while focusing on the specifics of the query.

$$PC(Q, C, P, K) = \ top - \ f_{pc}\big(Q, c_j, p_j\big),\ (c_j, p_j) \in C \times P \tag{2}$$

The combined set of results from both retrieval meth- ods is represented as R, where $R = SW\ (Q, D, K)\ \cup PC(Q, D, K)$. The result set R undergoes a re-ranking process, (3), to further refine relevance and contextual alignment.

$$R\ rank(R) = sort\ f_{rr}(r),\ r \in R \tag{3}$$

By considering the fine-grained relevance of the embeddings alongside the contextual richness provided by matching the child and parent chunks, the re-ranking ensures that the selections fed into the LLM, as can be seen in (4), for answer generation are not only relevant but also contextually coherent. The outcome is an enhanced response that leverages both the specificity of detailed information from the embeddings and the comprehensive background provided by the parent chunks.

$$A = LLM(Q, R_{rank(R)}) \tag{4}$$

Where A in (4) is the enhanced response generated by the LLM.

III. Experiments and Results

This section covers the dataset introduction, experimental settings, benchmark metrics, results presentation, and comparison of our method against state-of-the-art approaches.

A. Dataset

The Paul Graham Essay Dataset is designed for the evaluation and training of RAG models [18]. This dataset is

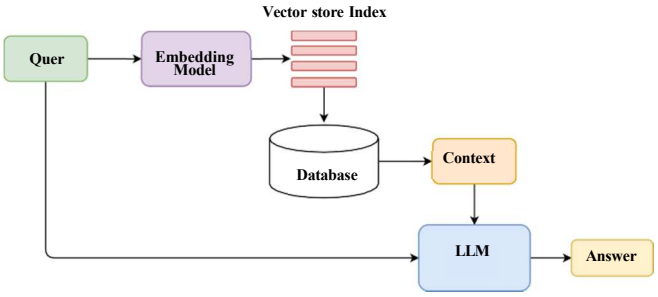

Fig. 1. Naive RAG pipeline is processed by an embed- ding model to transform the query into a vector represen- tation.

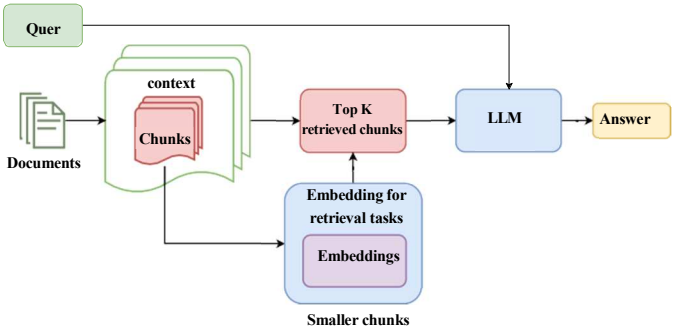

Fig. 2. Sentence-Window RAG pipeline, which focuses on smaller units of text within a larger context for infor- mation retrieval.

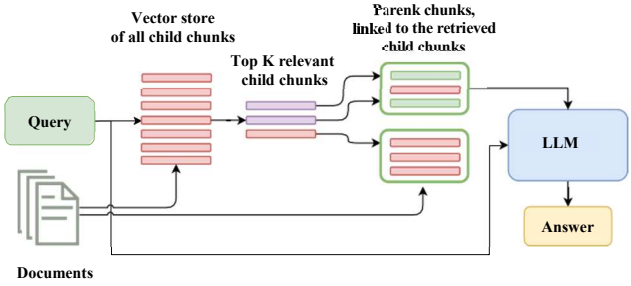

Fig. 3. Parent-Child RAG pipeline, which contains parent chunks and child chunks.

structured to support the development and evaluation of RAG models through various fields, including:

Reference Contexts: Passages from the essays that provide context for the queries.

Reference Answer: The answers to the queries, derived directly from the essay content.

Reference Answer By: The entity that provided the answer, typically annotated as generated by GPT4 LLM.

Query By: The source of the query, also typically annotated as generated by GPT4 LLM.

B. Experimental Settings

For the Naive RAG, Sentence-Window, and Parent- Child methods, the top k is set to 3. This means that these methods retrieve the three most relevant contexts for processing. In the proposed method, the re-ranking module processes the outputs from each Sentence-Window and Parent-Child module, selecting the three contexts with the highest scores, and then feeds these top contexts to the LLM. The model used for the LLM across all methods is GPT-3.5. Each method is repeated

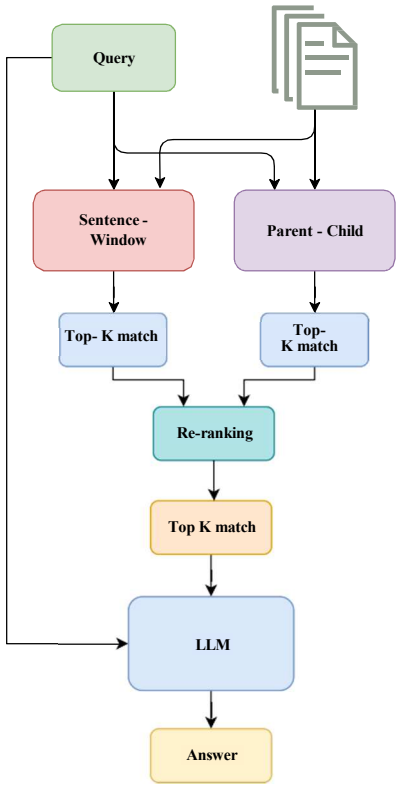

Fig. 4. Proposed method pipeline, including both Parent-Child and Sentence-Window methods, with re- ranking modules.

100 times, and mean and standard deviation of each metric for all methods are reported in Table I.

C. Metrics

The development of RAG models has become increas- ingly important in the NLP field, emphasizing the neces- sity for effective evaluation methodologies. Traditional approaches to RAG model evaluation have concentrated on performance in specific tasks, employing metrics such as accuracy for fact-checking or Exact Match (EM) and F1 scores for question answering [19], [20], [21]. However, there is an expanding interest in a broader assessment of RAG models, moving beyond merely task-specific mea- sures [12]. Recent research initiatives have underscored the significance of three primary quality scores for evaluat- ing RAG models: correctness, relevancy, and faithfulness. These metrics are designed to gauge the extent to which RAG models generate responses that are accurate, related to the given query, and true to the source material[22], [23].

1) Mean Correctness Score: The correctness evaluation metric quantitatively assesses the accuracy and relevance of responses generated by question-answering systems in relation to a given query and a reference answer[24], [25]. Calculation: Let Q represent the user query, R the response generated by the system, and Aref the optional reference answer. The evaluation is conducted using a scoring function $S : (Q, R, Aref) \rightarrow [1, 5]$, which assigns a score reflecting the correctness and relevance of R with respect to Q and Aref [18]. The scoring criteria are defined as follows:

- A score of 1 indicates that the response R is not relevant to the query Q.

  Scores between 2 and 3 are assigned if R is relevant but contains inaccuracies or is partially incorrect.

TABLE I. EVALUATION SCORES OF DIFFERENT RAG MODELS, INCLUDING BOTH MEAN AND STANDARD DEVIATION (STD). THE BEST RESULTS ARE MARKED IN BOLD STYLE.

Model	Correctness Score (±std)	Relevancy Score (±std)	Faithfulness Score (±std)
Naive RAG [13]	3.97 (±0.017)	0.75 (±0.008)	0.98 (±0.005)
Sentence-Window [14]	4.05 (±0.062)	0.85 (±0.043)	0.99 (±0.020)
Parent-Child [18]	4.06 (±0.069)	0.82 (±0.076)	0.99 (±0.01)
Proposed Method	**4.20 (±0.015)**	**0.86 (±0.037)**	**1 (±0.00)**

- A score between 4 and 5 is given when R is both relevant and accurate, with 5 indicating exceptional correctness and relevance.

Operational Details*:* The evaluation leverages a LLM to interpret and score the response based on predefined templates. These templates guide the LLM in evaluating R against Q and Aref. The process involves constructing an evaluative prompt incorporating Q, R, and Aref, which is then processed by the LLM to output a numerical score and qualitative feedback. This approach allows for a nuanced assessment that accounts not only for factual accuracy but also for the relevance of the response to the query [26].

2) ***Mean Relevancy Score:*** The relevancy evaluation Metric is designed to assess the appropriateness and alignment of a response generated by a language model in relation to a specific query and its associated context information [27].

Calculation: Given a query Q, a sequence of context documents $C = \{c1, c2, \ldots, cn\}$, and a generated response R, the relevancy evaluation employs a binary scoring function $E : (Q, R, C) \rightarrow \{0, 1\}$. The scoring function assesses whether the information presented in R is relevant and aligned with the context provided by C and directly answers Q [18]. The evaluation is operationalized through the following steps:

- A score of 1 (Yes) is assigned if R is deemed relevant and in line with the context information C for the query Q.
- A score of 0 (No) is assigned if R fails to align with the context information or does not adequately address Q.

Operational Details: The evaluation process utilizes a LLM along with predefined evaluation and refinement prompt templates to guide the assessment of R's relevancy. This involves constructing a prompt that encapsulates Q, R, and C and querying the LLM to ascertain the relevancy of the response.

3) ***Mean Faithfulness Score:*** The faithfulness metric serves as a quantitative measure to evaluate whether the responses generated by RAG systems are accurately grounded in the provided context documents [28], [29].

Calculation: The evaluation framework is defined by a set of context documents $C = \{c1, c2, \ldots, cn\}$ and the generated response R. We introduce an evaluation function $F : (R, C) \rightarrow \{0, 1\}$, where F outputs 1 if R is supported by C, indicating faithfulness, and 0 otherwise. This assessment leverages the LLM's ability to determine if R is consistent with the information contained within C, utilizing predefined evaluation (T_{eval}) and refinement (T_{refine}) templates [18].

Operational Details: This process utilizes a SummaryIndex and a query engine to efficiently evaluate the faithfulness of R with respect to C, harnessing the LLM's capabilities for deep textual understanding and inference. By doing so, the faithfulness metric ensures that the content generated by the RAG system is not only relevant but also factually accurate and grounded in the provided contexts [26].

D. Result Analysis and Performance Comparison

The proposed method demonstrates superior performance across all evaluated metrics when compared to the other models: naive RAG, Parent-child, and Sentence window as seen in Table I.

Mean Correctness Score: The Hybrid model achieves the highest correctness score of 4.20, indicating its superior ability to generate correct and accurate responses. This score is a direct measure of the model's effectiveness in understanding and processing the input to produce valid outputs. The closest competitor, the Parent-child model, scores slightly lower at 4.06, suggesting that while it is effective, it does not match the precision offered by the Hybrid model.

Mean Relevancy Score: With a score of 0.86, the Hybrid model also leads in producing responses that are most relevant to the queries or prompts given. This metric is crucial for assessing whether the model's outputs are not just correct but also on-topic. The Sentence-Window model follows closely with a score of 0.85, showing that it too can generate relevant responses, slightly less effectively than the Hybrid model.

***Mean Faithfulness Score*:** The Hybrid model achieves a perfect faithfulness score of 1, indicating that its responses are entirely faithful to the input data, with no instances of misrepresentation or factual inaccuracies. This score sets a benchmark for reliability, showing that the Hybrid model can be trusted to generate outputs that accurately reflect the source material. Both the Parent- child and Sentence window models score slightly lower at 0.99, which is still commendable but denotes a minor margin for error.

Performance Comparison: The comparative analy-sis of the models underscores the effectiveness of the Hy- brid approach in retrieval augmented generation for LLMs. While the naive RAG model provides a baseline with decent performance across metrics, it is noticeably out- performed by the more sophisticated models. The Parent- Child and Sentence-Window models offer improvements, particularly in relevancy and faithfulness, indicating that their specific strategies for handling data retrieval and integration yield better alignment with the input data.

However, the Hybrid model leading scores across all metrics suggest that its approach to combining retrieval and generation techniques is most effective. This model benefits from a more nuanced integration of external knowledge and

context, which enhances the correctness, relevancy, and faithfulness of its outputs. Its performance demonstrates the potential of hybrid models to signifi- cantly improve the quality and reliability of generated text in LLMs.

IV. Conclusion

This paper introduces a hybrid approach that combines two advanced RAG models with a re-ranking module. This approach enhances LLMs query responses by inte- grating additional information or domain-specific knowl- edge that was not available during their initial training phase. According to subsection III-D, the proposed hybrid RAG method achieved a correctness score of 4.20, a relevancy score of 0.86, and a faithfulness score of 1 on the benchmark dataset, surpassing previous state-of-the- art methods. Future research directions could explore further refinement of the re-ranking mechanisms and the integration of additional multimodal knowledge sources for developing multimodal hybrid RAGs.

Acknowledgment

We extend our gratitude to the Adak Vira Iranian Rahjoo (Avir) company for their invaluable assistance with this study.

References

[1] P. Omrani, Z. Ebrahimian, R. Toosi, M. A. Akhaee, Bilingual covid-19 fake news detection based on lda topic modeling and bert transformer, in: 2023 6th International Conference on Pattern Recognition and Image Analysis (IPRIA), IEEE, 2023,

[2] pp. 01–06.

[3] M. S. Jahan, M. Oussalah, A systematic review of hate speech automatic detection using natural language processing., Neurocomputing (2023) 126232.

[4] M. Zulqarnain, R. Ghazali, M. Aamir, Y. M. M. Hassim, An efficient two-state gru based on feature attention mechanism for sentiment analysis, Multimedia Tools and Applications 83 (1) (2024) 3085–3110.

[5] I. Ortiz-Garces, J. Govea, R. O. Andrade, W. Villegas-Ch, Optimizing chatbot effectiveness through advanced syntactic analysis: A comprehensive study in natural language processing, Applied Sciences 14 (5) (2024) 1737.

[6] L. Floridi, M. Chiriatti, Gpt-3: Its nature, scope, limits, and consequences, Minds and Machines 30 (2020) 681–694.

[7] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux,

[8] T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[9] M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili, et al., A survey on large language models: Applications, challenges, limitations, and practical usage, Authorea Preprints (2023).

[10] B. Li, T. Meng, X. Shi, J. Zhai, T. Ruan, Meddm: Llm-executable clinical guidance tree for clinical decision-making, arXiv preprint arXiv:2312.02441 (2023).

[11] J. Zeng, R. Huang, W. Malik, L. Yin, B. Babic, D. Shacham,

[12] X. Yan, J. Yang, Q. He, Large language models for social networks: Applications, challenges, and solutions, arXiv preprint arXiv:2401.02575 (2024).

[13] F. Trad, A. Chehab, Prompt engineering or fine-tuning? a case study on phishing detection with large language models, Machine Learning and Knowledge Extraction 6 (1) (2024) 367–

[14] 384.

[15] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang,

[16] L. Wang, W. Chen, Lora: Low-rank adaptation of large language models, arXiv preprint arXiv:2106.09685 (2021).

[17] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun,

[18] H. Wang, Retrieval-augmented generation for large language models: A survey, arXiv preprint arXiv:2312.10997 (2023).

[19] S. Yih, Retrieval-augmented generation for knowledge-intensive nlp tasks (2020).

[20] P. Finardi, L. Avila, R. Castaldoni, P. Gengo, C. Larcher,

[21] M. Piau, P. Costa, V. Caridá, The chronicles of rag: The retriever, the chunk and the generator, arXiv preprint arXiv:2401.07883 (2024).

[22] X. Cheng, D. Luo, X. Chen, L. Liu, D. Zhao, R. Yan, Lift yourself up: Retrieval-augmented text generation with self- memory, Advances in Neural Information Processing Systems 36 (2024).

[23] S. Siriwardhana, R. Weerasekera, E. Wen, T. Kaluarachchi,

[24] R. Rana, S. Nanayakkara, Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering, Transactions of the Association for Computational Linguistics 11 (2023) 1–17.

[25] W. Chen, H. Hu, X. Chen, P. Verga, W. W. Cohen, Murag: Multimodal retrieval-augmented generator for open question answering over images and text, arXiv preprint arXiv:2210.02928 (2022).

[26] J. Liu, Llamaindex (11 2022). doi:10.5281/zenodo.1234.

[27] URL https://github.com/jerryjliu/llama$_i$*ndex*

[28] B. Wang, W. Ping, L. McAfee, P. Xu, B. Li, M. Shoeybi,

[29] B. Catanzaro, Instructretro: Instruction tuning post retrieval-augmented pretraining, arXiv preprint arXiv:2310.07713 (2023).

[30] Z. Feng, X. Feng, D. Zhao, M. Yang, B. Qin, Retrieval- generation synergy augmented large language models, arXiv preprint arXiv:2310.05149 (2023).

[31] X. Ma, Y. Gong, P. He, H. Zhao, N. Duan, Query rewriting for retrieval-augmented large language models, arXiv preprint arXiv:2305.14283 (2023).

[32] J. Saad-Falcon, O. Khattab, C. Potts, M. Zaharia, Ares: An automated evaluation framework for retrieval-augmented generation systems, arXiv preprint arXiv:2311.09476 (2023).

[33] S. Es, J. James, L. Espinosa-Anke, S. Schockaert, Ragas: Automated evaluation of retrieval augmented generation, arXiv preprint arXiv:2309.15217 (2023).

[34] E. Kamalloo, N. Dziri, C. L. Clarke, D. Rafiei, Evaluating open-domain question answering in the era of large language models, arXiv preprint arXiv:2305.06984 (2023).

[35] J. Bulian, C. Buck, W. Gajewski, B. Boerschinger, T. Schuster, Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation, arXiv preprint arXiv:2202.07654 (2022).

[36] V. Adlakha, P. BehnamGhader, X. H. Lu, N. Meade, S. Reddy, Evaluating correctness and faithfulness of instruction-following models for question answering, arXiv preprint arXiv:2307.16877 (2023).

[37] W. Sakata, T. Shibata, R. Tanaka, S. Kurohashi, Faq retrieval using query-question similarity and bert-based query-answer relevance, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 1113–1116.

[38] N. Dziri, E. Kamalloo, S. Milton, O. Zaiane, M. Yu, E. M. Ponti,

[39] S. Reddy, Faithdial: A faithful benchmark for information- seeking dialogue, Transactions of the Association for Compu- tational Linguistics 10 (2022) 1473–1490.

[40] V. Adlakha, S. Dhuliawala, K. Suleman, H. de Vries, S. Reddy, Topiocqa: Open-domain conversational question answering with topic switching, Transactions of the Association for Com- putational Linguistics 10 (2022) 468–483.