Assessing Support for the TREC 2024 RAG Track: A Large-Scale Comparative Study of LLM and Human Evaluations

Nandan Thakur
nandan.thakur@uwaterloo.ca
University of Waterloo
Waterloo, Canada

Ronak Pradeep
rpradeep@uwaterloo.ca
University of Waterloo
Waterloo, Canada

Shivani Upadhyay
sjupadhyay@uwaterloo.ca
University of Waterloo
Waterloo, Canada

Daniel Campos
daniel.campos@snowflake.com
Snowflake
San Mateo, United States

Nick Craswell
nickcr@microsoft.com
Microsoft
Seattle, United States

Ian Soboroff
ian.soboroff@nist.gov
NIST
Gaithersburg, United States

Hoa Trang Dang
hoa.dang@nist.gov
NIST
Gaithersburg, United States

Jimmy Lin
jimmylin@uwaterloo.ca
University of Waterloo
Waterloo, Canada

Abstract

Retrieval-augmented generation (RAG) enables large language models (LLMs) to generate answers with citations from source documents containing "ground truth". A crucial factor in RAG evaluation is "support", or whether the information in the cited documents supports the answer. We conducted a comparative study of submissions to the TREC 2024 RAG Track, evaluating an automatic LLM judge (GPT-4o) against human judges for support assessment. We considered two conditions: (1) fully manual assessments from scratch and (2) manual assessments with post-editing of LLM predictions. Our results indicate good agreement between human and GPT-4o predictions. Further analysis of the disagreements shows that an independent human judge correlates better with GPT-4o than a human judge, suggesting that LLM judges can be a reliable alternative for support assessment. We provide a qualitative analysis of human and GPT-4o errors to help guide future evaluations.

CCS Concepts

• **Information systems** → **Relevance assessment**.

Keywords

Retrieval-Augmented Generation; Support Evaluation; LLM Judge.

ACM Reference Format:
Nandan Thakur, Ronak Pradeep, Shivani Upadhyay, Daniel Campos, Nick Craswell, Ian Soboroff, Hoa Trang Dang, and Jimmy Lin. 2025. Assessing Support for the TREC 2024 RAG Track: A Large-Scale Comparative Study of LLM and Human Evaluations. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3726302.3730165

1 Introduction

Retrieval-Augmented Generation (RAG) has recently gained popularity in both academic and industrial settings (e.g., Bing Search [14] and popular frameworks like LangChain [2]). In RAG, large language models (LLMs) generate answers to user queries that include citations to source documents as necessary [1, 8, 9, 11]. RAG systems improve factuality and verifiability, reducing hallucinations observed in "closed-book" LLM generation [7, 10, 11, 13, 21].

Document-level citations for supporting facts in LLM-generated answers are integral to any deployed RAG system. Therefore, support evaluation assesses whether a RAG answer factually supports the information present in the cited documents, which is crucial for evaluating the quality of a RAG system. Prior work on support evaluation in the RAG literature [3, 6, 12, 19, 20, 22, 23] relies on an *automatic* judge, i.e., an LLM as a proxy judge. However, it is unknown whether an LLM judge can potentially replace a human judge for support evaluation.

This paper examines results from the TREC 2024 RAG Track, assessing 45 participant systems on 36 information-based queries. A sample query and answer are shown in Table 1. We conducted a large-scale comparative study between human and LLM judges using resources provided by the National Institute of Standards and Technology (NIST) to better understand whether support assessment can be automated. Unique to the TREC setup, we contrast our automatic judgment process using a strong LLM judge (like GPT-4o) against a manual process under two conditions: (1) *manual from scratch*, where human annotators perform assessments from scratch and (2) *manual with post-editing*, where human annotators are shown GPT-4o predictions during the evaluation process.

In this paper, we focus exclusively on support, i.e., whether the information in an answer sentence is supported by the cited documents, which we consider as the "ground truth". Our experimental results indicate that GPT-4o and human judgments perfectly match 56% of the time in the manual from-scratch condition, increasing to 72% in the manual with post-editing condition. These results show promise in using LLM judges for support assessment in both conditions. We measured support of a system's overall answer in terms of two metrics: weighted precision and weighted recall, where precision penalizes overcitation, and recall penalizes undercitation. We observe a high correlation at the run level (above 0.79 Kendall τ) between GPT-4o and human judges, providing evidence that LLMs can potentially replace human judges for support evaluation.

In addition, to better understand the discrepancies between GPT-4o and human judges, we conducted an unbiased disagreement study with an independent human judge who carefully re-assessed 537 randomly sampled pairs, including both assessment conditions. Our results surprisingly show that the independent judge agrees

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730165

Topic: how taylor swift's age affects her relationships
Answer: Swift's relationships have frequently involved notable age gaps, which have sometimes led to complications [ ]. For instance, her relationship with John Mayer, who was 11 years her senior, reportedly strained due to the age difference and Mayer's reputation as a playboy, leading to Swift feeling taken advantage of [0, 3, 4]. This relationship inspired her song "Dear John," reflecting her emotional turmoil [3, 4]. [. . .] As she matured, her understanding of relationships evolved, making her less naive and more discerning in her romantic choices [8]. In summary, Taylor Swift's age has played a crucial role in shaping her relationships, influencing both the dynamics and outcomes [8, 1].

Answer Sentence: This relationship inspired her song "Dear John," reflecting her emotional turmoil.

Passage ID [3]: `msmarco_v2.1_doc_35_202251892#8_427548986`
Title: Timeline Of Taylor Swift's Age-Inappropriate Romances | Business Insider
Text: [. . .] 2010: Taylor Swift, 21, & John Mayer, 321 / 12 And then the inappropriateness of Swift's dating habits peaked [. . .] and then the heartbroken young Swift penned the song 'Dear John' about the break up. Earlier this year, Mayer admitted that he felt 'humiliated' when he heard the song, but Swift refuses to admit it's about him, telling Glamour magazine it was 'presumptuous' of him to think the song was about him.

Human Judge: Full Support **GPT-4o Judge:** Full Support

Answer Sentence: As she matured, her understanding of relationships evolved, making her less naive and more discerning in her romantic choices.

Passage ID [8]: `msmarco_v2.1_doc_48_737500982#1_1325021022`
Title: What Went Wrong With Jake Gyllenhaal And Taylor Swift?
Text: Taylor Swift and Jake Gyllenhaal dated from October to December 2010. [. . .] "He said he could feel the age difference," a source told Us Weekly. [. . .] "When Jake broke her heart, she was so inexperienced she didn't know how to deal with it. She wasn't used to all the head games and the lies but now she's way less naive."

Human Judge: No Support **GPT-4o Judge:** Partial Support

Table 1: Examples of support evaluation with GPT-4o and human judges for the Taylor Swift topic. The citations in the answer and the fragment of the passage that supports the answer sentence are highlighted.

better with GPT-4o than the human judge (e.g., Cohen's κ of 0.27 vs. 0.07). For a much more thorough analysis of support evaluation results, please refer to Thakur et al. [21]. Finally, we discuss annotation errors to help improve future iterations of support evaluation.

2 Background and Related Work

In our work, we evaluate support at the sentence level in the answer as defined in Thakur et al. [21]. We assume an answer r is segmented into n sentences, $r = \{a_1, \cdots, a_n\}$, where each answer sentence a_i can contain a maximum of m document citations, $a_i = \{d_i, \cdots, d_m\}$, each of which is a document drawn from the corpus. Support is calculated as the function $f(a_i, d_j) = s_{i,j}$ where f can be a human or LLM judge that generates a scalar value $s_{i,j}$, indicating the extent that the cited document d_j provides support to sentence a_i. A few examples of support evaluation are shown in Table 1.

Previous work on support evaluation in RAG used different automatic judges: examples include an natural language inference (NLI) model [7], LLM with prompting [6], or even fine-tuned custom LLMs [20] as the automatic judge. Wu et al. [22] evaluated the tug of war between an LLM's internal prior over supporting wrong context information. Similar to our formulation, Ming et al. [15] provided an evaluation benchmark consisting of academic question answering (QA) datasets with human validation, and Liu et al. [12] evaluated the quality of proprietary search engine outputs with crowdsourced human judges.

	Condition	#Topics	#Annotations	Support level FS	PS	NS
(1a)	Manual from scratch (Human)	22	6,742	2,752	1,652	2,338
(1b)	Automatic (GPT-4o)	22	6,742	3,110	2,421	1,211
(2a)	Manual with post-editing (Human)	14	4,165	1,812	1,076	1,277
(2b)	Automatic (GPT-4o)	14	4,165	2,045	1,330	790

Table 2: Descriptive statistics for support judgments for the (1) manual from-scratch condition and (2) manual with post-editing condition for 45 participant submissions on 36 topics.

3 Track Description & Assessment

3.1 TREC 2024 RAG Track

The context for this work is the TREC 2024 RAG Track[1] [17, 21], where many teams participated, organized by the Text Retrieval Conference (TREC). Therefore, our human and LLM judges were exposed to multiple answers and cited documents during the support evaluation. The track required that answers generated for topics be segmented into sentences, and that each sentence is associated with citations to passages from the corpus (e.g., Table 1).

Passage collection. The MS MARCO V2.1 segment collection contains 113,520,750 passages, derived from a deduplicated version of the MS MARCO V2 document collection [4] by removing near-duplicate documents using locality-sensitive hashing (LSH) with MinHash and 9-gram shingles.

Topic collection. For the TREC 2024 RAG Track topics (queries), we leveraged a fresh scrape of Bing Search logs containing non-factoid queries that are multifaceted and subjective, warranting RAG systems to provide long-form answers [17, 18].

3.2 Support Assessment

Consistent with previous support evaluations in RAG [7, 12], we used a three-level grade, with the following associated descriptions[2] for each support level:

FS **Full Support**: All of the information in the answer sentence is factually consistent with and supported by the cited passage.
PS **Partial Support**: Some of the information in the answer sentence is factually consistent with and supported by the cited passage, but other parts of the sentence are not supported.
NS **No Support**: The cited passage is completely irrelevant and does not support any part of the answer sentence.

Next, to evaluate the quality of LLM judges in contrast to human judges, we conducted our support assessment with human judges under two conditions: (1) manual from scratch and (2) manual with post-editing. We describe both conditions below:

Manual from scratch. A human judge is provided the answer sentence and the cited passage, who assigns one of the labels above.

Manual with post-editing. Same as above, except that the human judge is additionally given the label from the LLM judge.

For automatic labeling, we utilized GPT-4o as an automatic judge. We ran inference using the Microsoft Azure API [16]. The GPT-4o judge is presented with each sentence and its cited passage and asked to determine the support label without any explanation

[1] TREC 2024 RAG Track website: https://trec-rag.github.io
[2] An edge case is a sentence with zero citations: We automatically consider the support assessment to be "no support", as the sentence does not cite any retrieved passage.

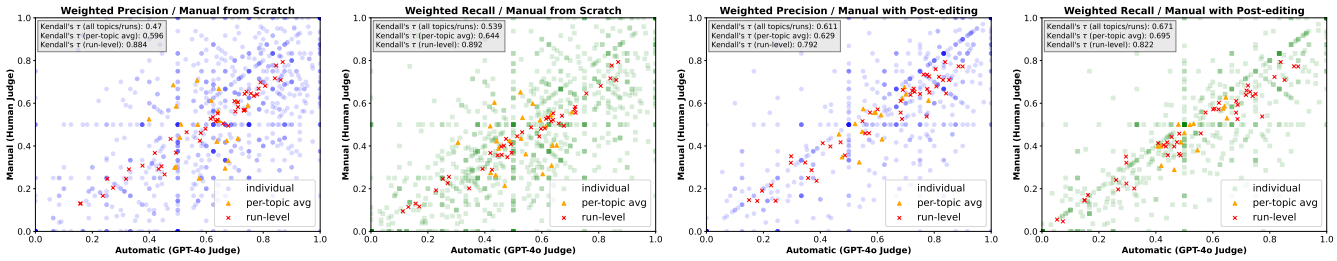

Figure 1: Correlations between weighted precision and recall scores from human and GPT-4o judges for the manual from-scratch and manual with post-editing conditions. Red markers show run-level scores, yellow triangles show per-topic averages, and blue dots or green boxes show all individual topic/run combinations. Each plot contains rank correlations showing Kendall's τ.

(full support, partial support, or no support). The prompt used for support evaluation with GPT-4o is described in Thakur et al. [21].

3.3 Computational Cost & Evaluation Tradeoffs

In the TREC 2024 RAG Track, we allowed participants to provide citations for up to 20 passages per answer sentence. To judge each sentence and its cited passage, our protocol requires a human judge to read the answer sentence and a relatively long text passage (typically, 500–1000 characters). Thus, conducting an exhaustive evaluation of all cited passages for every answer sentence across multiple participants was not feasible given our budget.

Therefore, we had to choose between sparse and dense annotations. Dense annotations would provide fewer judged topics, but each answer sentence would be evaluated against k cited passages. On the other hand, sparse annotations would provide higher diversity in judged topics. We opted for sparse annotations to achieve more judged topics, at the cost of judging fewer cited passages for each answer sentence. We fixed both the human and GPT-4o judge to evaluate only the first cited passage of every answer sentence for all participants. NIST provided the resources to perform human evaluations based on the guidance of the track organizers (i.e., us). NIST first trained every human judge to understand the task, and then each human judge evaluated each topic sequentially.

3.4 Support Evaluation Metrics

Support can be evaluated across two dimensions, similar to Liu et al. [12]: (1) *weighted precision*, accounting for how many correct passage citations are present in the generated answer, and (2) *weighted recall*, accounting for how many sentences in the answer are supported by passage citations We define both metrics below:

Weighted precision. This metric measures the weighted proportion of citations that support each answer sentence. We assign a weight to $s(a_i, d_j)$ of 1.0 to Full Support (FS), 0.5 to Partial Support (PS), and 0 to No Support (NS) for the answer sentence and passage.

Weighted recall. This metric measures the weighted proportion of answer sentences that are supported by their cited passages. We assign the same weights as defined above in weighted precision.

4 Experimental Results

For the TREC 2024 RAG Track, human judges completed judgments for 36 topics from 45 participant runs, sparsely annotated: 6,742 annotations on 22 topics in the manual from-scratch condition and 4,165 annotations on 14 topics in the manual with post-editing condition (detailed statistics are provided in Table 2).

4.1 Weighted Precision and Recall

Figure 1 shows scatter plots of weighted precision and recall scores. Run-level scores (×) are strongly correlated (above 0.79 Kendall's τ) between GPT-4o and human annotations. Per-topic averages (△) vary on both axes, where certain topics achieved a higher weighted precision and recall score than humans over GPT-4o, and vice versa. Individual participant scores (○ or □) showed a high variance in both weighted precision and recall scores. This was likely due to the mismatch of human annotators preferring "no support", whereas GPT-4o prefers "partial support". Overall, we observed high scores in the bottom right triangle, which showed that humans took a more conservative approach and provided lower levels of support.

4.2 Confusion Matrices

Next, to better understand how often the GPT-4o judge agrees with the human judges, we plot the confusion matrices in Figure 2.

Manual from-scratch condition. For 56% (13.7% + 11.9% + 30.4%), GPT-4o and the human judge perfectly agreed on their support judgment on 22 topics. Both "full support" and "no support" categories have higher percentages (30.4% and 13.7%), showing that humans and GPT-4o as judges agreed more on both ends of the spectrum. For 15.1%, the GPT-4o judge considered an annotation as "partial support", which the human judge annotated as "no support". An important observation was that GPT-4o was more likely to provide a higher support label than the human judge (upper right triangle has a higher combined percentage over the lower left triangle).

Manual with post-editing condition. From the previous condition, we saw an increase in perfect agreement rise to 72.1% (15.9% + 18.7% + 37.5%) on 14 topics annotated with post-editing GPT-4o labels. Therefore, sentences and cited passages with "partial support" that led to disagreements earlier in the manual from-scratch condition were now reduced. In this condition, human judges were likely to agree with GPT-4o unless it is a mistake, i.e., when the GPT-4o judge considered an annotation to be "full support" and the human judge considered it to be "no support" (increased to 6.3% from 5.9% in the manual from-scratch condition).

5 Annotator Disagreements

In the experiments reported in Section 4, we observed frequent disagreements between the human and GPT-4o judge. To further

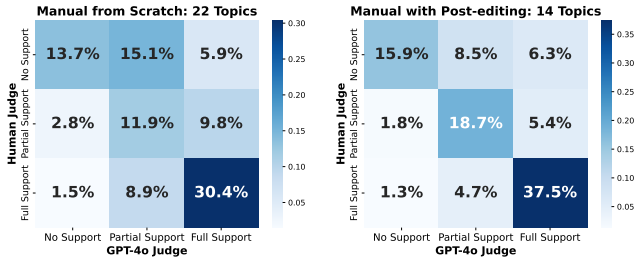

Figure 2: Confusion matrices comparing predictions from human and GPT-4o judges for the manual from-scratch condition (left) and the manual with post-editing condition (right).

	Cohen's Kappa	Manual from scratch		Manual with post-editing	
		GPT-4o	Human	GPT-4o	Human
(1)	Expert (human)	**0.29**	-0.03	**0.27**	0.07
(2)	LLAMA-3.1 (405B)	**0.60**	-0.20	**0.46**	-0.06

Table 3: Inter-annotator correlation score (Cohen's Kappa) for an unbiased study on disagreements between GPT-4o and human annotators on both manual conditions.

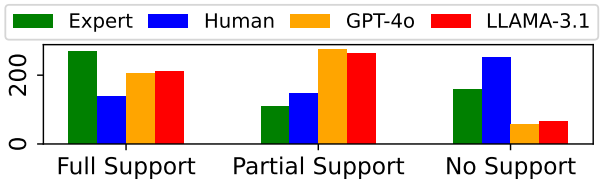

Figure 3: Support label prediction by different judges for each category (FS, PS, NS) in the disagreement analysis.

study this, we performed unbiased annotations from scratch by carefully re-assessing the support judgment of randomly sampled disagreements between the human and GPT-4o judge, with an independent human judge and another LLM judge using LLAMA-3.1 405B [5] (with the same prompt as GPT-4o). We randomly sampled 15 disagreement pairs per topic, re-assessing 537 sentences and their first cited passages, including both assessment conditions: (1) manual from scratch and (2) manual with post-editing.

Results. As shown in Table 3, we interestingly found the independent human judge to be *better correlated* with GPT-4o than the human judge provided by NIST (Cohen's κ of 0.29 and 0.27 versus -0.03 and 0.07) in the manual from-scratch condition. The independent judge fully matched 31% of the time with the human judge and 51% of the time with the GPT-4o judge. Similarly, in the manual with post-editing condition, the independent judge fully matched 37% of the time with the human judge and 52% of the time with the GPT-4o judge. LLAMA-3.1 405B had a stronger correlation with another LLM (GPT-4o) over human judges (Cohen's κ of 0.60 and 0.46 versus -0.20 and -0.06), demonstrating the high likelihood of different LLMs providing similar prediction labels.

From the label distributions shown in Figure 3, we observed that both LLMs (LLAMA-3.1 405B and GPT-4o) labeled about 49–51% of sentences and their cited passage as "partial support", whereas the human judge labeled 47% of the sentences as "no support". The independent judge labeled 50% of the sentences as "full support".

Qualitative analysis. We further assessed examples qualitatively to understand failure cases, for example, when a human or GPT-4o judge makes mistakes during support evaluation. Overall, we summarize a few of the following errors made by GPT-4o: (1) GPT-4o can confuse words or phrases with similar meanings; for example, it is unable to distinguish between police and security specialists. (2) GPT-4o can miss out on evaluating the whole sentence (especially information present at the end of the sentence), biasing towards the "full support" label, and (3) GPT-4o can label "partial support" if the theme in the answer sentence is similar, but the passage does not support any text present in the answer sentence, i.e., "no support".

On the other hand, human judges make mistakes due to not reading the passages carefully. In some cases, answer sentences directly stated in the middle or at the end of a passage, or mentioned in parts of the passage, were surprisingly unnoticed by a human judge, causing judgment "no support" instead of "full support". We also observe that a human judge occasionally labels an answer sentence as "full support" even though the passage doesn't provide any support information. We suspect this is due to an inherent bias relying on the human judge's memory or understanding of the topic, instead of strictly relying on the actual passage text.

6 Conclusion

In this work, we evaluated support in RAG answers by analyzing submissions from the TREC 2024 RAG Track in a large-scale comparative study involving both humans and LLMs as judges. We critiqued and evaluated strong LLM judges, like GPT-4o, against human annotators for support assessment. Our results show a high level of agreement between GPT-4o and human judgments. We observe that disagreements between humans and LLMs mainly occur for sentence–passage pairs indicating partial support, i.e., in the middle of the support evaluation spectrum.

To better understand these disagreements, we conducted an unbiased evaluation by carefully re-assessing judgments with an independent human judge and a different LLM. Interestingly, in cases of disagreements, both the independent human judge and the LLAMA-3.1 judge agreed more with the GPT-4o judge than with the human judge, providing evidence for widely divergent opinions and perhaps the veracity of using LLMs for support evaluation. Further research could explore the nuances of disagreements between human and LLM judges and investigate limitations of both humans and LLMs to improve future iterations of support assessment.

Acknowledgments

We'd like to thank the annotator team at NIST and Corby Rosset for providing the test queries. This research was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada. Additional funding is provided by Snowflake, Microsoft via the Accelerating Foundation Models Research program, and a grant by the Korean Government (No. RS-2024-00457882, National AI Research Lab Project). Certain software or materials are identified in this paper in order to specify the experimental procedure adequately and is not intended to imply recommendation or endorsement of any product or service by NIST. These opinions, recommendations, findings, and conclusions do not necessarily reflect the views or policies of NIST or the United States Government.

References

[1] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2022. Improving Language Models by Retrieving from Trillions of Tokens. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research, Vol. 162)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (Eds.). PMLR, 2206–2240.

[2] Harrison Chase. 2022. *LangChain.*

[3] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking Large Language Models in Retrieval-Augmented Generation. *Proceedings of the AAAI Conference on Artificial Intelligence* 38, 16 (Mar. 2024), 17754–17762. doi:10.1609/aaai.v38i16.29728

[4] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2022. Overview of the TREC 2022 Deep Learning Track. In *Proceedings of the Thirty-First Text REtrieval Conference (TREC 2022)*. Gaithersburg, Maryland.

[5] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, and 520 others. 2024. The Llama 3 Herd of Models. *CoRR* abs/2407.21783 (2024). doi:10.48550/ARXIV.2407.21783 arXiv:2407.21783

[6] Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated Evaluation of Retrieval Augmented Generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, Nikolaos Aletras and Orphee De Clercq (Eds.). Association for Computational Linguistics, St. Julians, Malta, 150–158. https://aclanthology.org/2024.eacl-demo.16

[7] Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling Large Language Models to Generate Text with Citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 6465–6488.

[8] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval Augmented Language Model Pre-Training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 3929–3938.

[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, EACL 2021, Online, April 19 - 23, 2021*, Paola Merlo, Jörg Tiedemann, and Reut Tsarfaty (Eds.). Association for Computational Linguistics, 874–880.

[10] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through Memorization: Nearest Neighbor Language Models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[11] Patrick S. H. 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 *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.).

[12] Nelson Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating Verifiability in Generative Search Engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 7001–7025. doi:10.18653/v1/2023.findings-emnlp.467

[13] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics* 12 (2024), 157–173. doi:10.1162/tacl_a_00638

[14] Microsoft. 2023. *Reinventing search with a new AI-powered Microsoft Bing and Edge, your copilot for the web.*

[15] Yifei Ming, Senthil Purushwalkam, Shrey Pandit, Zixuan Ke, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. 2024. FaithEval: Can Your Language Model Stay Faithful to Context, Even If "The Moon is Made of Marshmallows". *CoRR* abs/2410.03727 (2024). doi:10.48550/ARXIV.2410.03727 arXiv:2410.03727

[16] OpenAI. 2024. *Hello GPT-4o.*

[17] Ronak Pradeep, Nandan Thakur, Sahel Sharifymoghaddam, Eric Zhang, Ryan Nguyen, Daniel Campos, Nick Craswell, and Jimmy Lin. 2025. Ragnarök: A Reusable RAG Framework and Baselines for TREC 2024 Retrieval-Augmented Generation Track. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part I* (Lucca, Italy). Springer-Verlag, Berlin, Heidelberg, 132–148. doi:10.1007/978-3-031-88708-6_9

[18] Corby Rosset, Ho-Lam Chung, Guanghui Qin, Ethan C. Chau, Zhuo Feng, Ahmed Awadallah, Jennifer Neville, and Nikhil Rao. 2024. Researchy Questions: A Dataset of Multi-Perspective, Decompositional Questions for LLM Web Agents. *CoRR* abs/2402.17896 (2024). doi:10.48550/ARXIV.2402.17896 arXiv:2402.17896

[19] Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, Zizhao Zhang, Binjie Wang, Jiarong Jiang, Tong He, Zhiguo Wang, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024. RAGChecker: A Fine-grained Framework for Diagnosing Retrieval-Augmented Generation. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (Eds.). http://papers.nips.cc/paper_files/paper/2024/hash/27245589131d17368cccdfa990cbf16e-Abstract-Datasets_and_Benchmarks_Track.html

[20] Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 338–354. doi:10.18653/v1/2024.naacl-long.20

[21] Nandan Thakur, Ronak Pradeep, Shivani Upadhyay, Daniel Campos, Nick Craswell, and Jimmy Lin. 2024. Support Evaluation for the TREC 2024 RAG Track: Comparing Human versus LLM Judges. *CoRR* abs/2504.15205 (2024). doi:10.48550/ARXIV.2504.15205 arXiv:2504.15205

[22] Kevin Wu, Eric Wu, and James Zou. 2024. How faithful are RAG models? Quantifying the tug-of-war between RAG and LLMs' internal prior. *CoRR* abs/2404.10198 (2024). doi:10.48550/ARXIV.2404.10198 arXiv:2404.10198

[23] Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2024. Evaluation of Retrieval-Augmented Generation: A Survey. *CoRR* abs/2405.07437 (2024). doi:10.48550/ARXIV.2405.07437 arXiv:2405.07437