Hypothetical Documents or Knowledge Leakage? Rethinking LLM-based Query Expansion

Yejun Yoon[♡] **Jaeyoon Jung**[♣◇] **Seunghyun Yoon**[♠] **Kunwoo Park**[♣♡]

[♡]Department of Intelligent Semiconductors, Soongsil University
[♣]School of AI Convergence, Soongsil University
[◇]MAUM AI Inc.
[♠]Adobe Research, USA

{yejun0382, jaeyoonskr}@soongsil.ac.kr, syoon@adobe.com, kunwoo.park@ssu.ac.kr

Abstract

Query expansion methods powered by large language models (LLMs) have demonstrated effectiveness in zero-shot retrieval tasks. These methods assume that LLMs can generate hypothetical documents that, when incorporated into a query vector, enhance the retrieval of real evidence. However, we challenge this assumption by investigating whether knowledge leakage in benchmarks contributes to the observed performance gains. Using fact verification as a testbed, we analyzed whether the generated documents contained information entailed by ground truth evidence and assessed their impact on performance. Our findings indicate that performance improvements occurred consistently only for claims whose generated documents included sentences entailed by ground truth evidence. This suggests that knowledge leakage may be present in these benchmarks, inflating the perceived performance of LLM-based query expansion methods, particularly in real-world scenarios that require retrieving niche or novel knowledge.

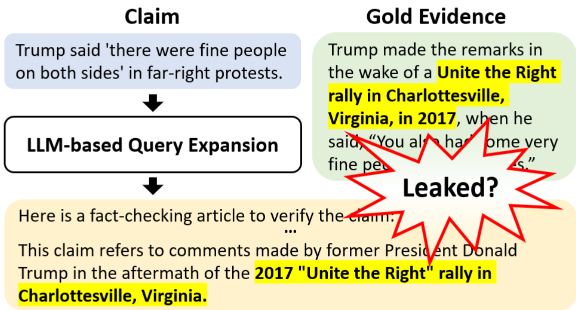

Figure 1: Illustration of potential knowledge leakage in LLM-based query expansion.

1 Introduction

Zero-shot retrieval aims to identify relevant documents without requiring any relevance supervision for training a retriever (Zhao et al., 2024). Because obtaining query-document pairs, such as MS-MARCO (Bajaj et al., 2016), for supervised training is challenging, developing zero-shot retrieval methods is both difficult and highly desirable for effectively addressing knowledge-intensive applications (Lewis et al., 2020), including question answering (Zhu et al., 2021) and fact verification (Guo et al., 2022).

Recent studies have leveraged the natural language generation capabilities of large language models (LLMs) to enhance the performance of zero-shot retrieval (Thakur et al., 2021). LLM-based query expansion (QE) uses LLMs to generate documents that extend a query (Jagerman et al., 2023; Lei et al., 2024; Mackie et al., 2023a). Approaches such as HyDE (Gao et al., 2023) and Query2doc (Wang et al., 2023), which have been widely adopted in recent research (Wang et al., 2024a; Chen et al., 2024b; Yoon et al., 2024), have achieved notable performance gains across various benchmarks without retriever parameter updates. These approaches prompt LLMs to generate documents that answer a question or verify a claim. Although these generated documents, referred to as *hypothetical* documents, may contain factual errors or hallucinations, it is assumed that incorporating them into a query can enhance retrieval of relevant *real* documents (Gao et al., 2023).

In this paper, we challenge the underlying assumption, as illustrated in Figure 1: *Do LLMs truly generate hypothetical documents, or are they merely reproducing what they already know?* LLMs are extensively pretrained on vast corpora, primarily collected from the web. As common retrieval targets, such as Wikipedia and web documents, are often included in these pretraining corpora (Groeneveld et al., 2024; Touvron et al., 2023; Du et al., 2022; Brown et al., 2020), many available LLMs may already contain knowledge relevant to a given query and retrieval targets, a phenomenon we refer to as *knowledge leakage*. If knowledge leakage occurs, it could lead to an overestimation of the effectiveness of QE methods in real-world scenarios, where generating documents for *unknown* knowledge is crucial for recent or niche queries.

To understand whether knowledge leakage exists and how it influences benchmark performance, this study examines LLM-generated documents using fact verification as a testbed. We analyze whether these documents contain sentences entailed by gold evidence and assess their impact on performance. Across experiments involving three benchmarks and seven LLMs, we observed a consistent trend: query expansion methods were effective only when LLM-generated documents included sentences entailed by gold evidence. This result implies that knowledge leakage within these benchmarks may artificially boost the observed effectiveness of LLM-based query expansion approaches.

2 Related Works

LLM-based Query Expansion QE has been explored as a means to enhance retrieval performance by enriching the initial query representation (Azad and Deepak, 2019). One widely studied approach is relevance feedback (Rocchio, 1971; Lavrenko and Croft, 2001; Amati and Van Rijsbergen, 2002), which leverages feedback signals to expand the query. Recent work has explored LLM's generative capabilities for QE (Zhu et al., 2023; Jagerman et al., 2023; Lei et al., 2024). Mackie et al. (2023b), for instance, introduced a method that utilizes LLM-generated documents as relevance feedback. Meanwhile, other researchers proposed HyDE (Gao et al., 2023) and Query2doc (Wang et al., 2023), which employs LLMs to generate hypothetical documents based on an initial query. Despite their simplicity, these methods have demonstrated substantial effectiveness across benchmarks for zero-shot retrieval and knowledge-intensive tasks (Wang et al., 2024a), including fact verification (Yoon et al., 2024).

Data Leakage and LLM Memorization Previous research has investigated various forms of data leakage in LLMs (Kandpal et al., 2023; Samuel et al., 2025; Deng et al., 2024; Xu et al., 2024a). One study used perplexity to detect potential data leakage, uncovering substantial instances of training or even test set misuse (Xu et al., 2024b). Deng et al. (2023) further examined data contamination by predicting masked tokens in test sets and found that GPT 3.5 could reconstruct missing portions of MMLU (Hendrycks et al., 2020) test instances with 57% accuracy. Other research has explored the boundaries of LLM knowledge (Yin et al., 2023; Dong et al., 2024; Burns et al., 2022; Kadavath et al., 2022), including a refusal-aware instruction tuning method that trains LLMs to reject uncertain questions (Zhang et al., 2024)—where an LLM is deemed uncertain if its generated response does not match the ground truth. Another study leveraged response consistency to estimate an LLM's confidence in its knowledge (Cheng et al., 2024). In this work, we apply NLI (MacCartney, 2009) to LLM-generated documents, referencing gold evidence, to examine what an LLM knows.

3 Methodology

3.1 Task and Dataset

Fact verification aims to predict the veracity label of a textual claim c. Depending on the dataset, the veracity label can fall into one of three or four categories[1]: *supported*, *refuted*, *not enough evidence*, or *conflicting evidence*.

The fact-verification task consists of two subtasks: evidence retrieval and verdict prediction. In evidence retrieval, a retrieval method $R(\cdot)$ identifies an evidence set $\tilde{E} = \{\tilde{e_1}, \cdots, \tilde{e_k}\}$ from a knowledge store K (e.g., Wikipedia), used to verify c. The performance of R is evaluated by comparing $\tilde{E}$ with $E = \{e_1, \cdots, e_l\}$, the gold evidence set. Verdict prediction then determine the veracity label of c based on $\tilde{E}$.

Dataset	# claim	# gold evidence per claim	# documents (knowledge source)
FEVER	6,666	1.66	5,416,536 (Wikipedia)
SciFact	693	1.8	5,183 (Paper abstracts)
AVeriTeC	3,563	2.77	2,623,538 (Web documents)

Table 1: Dataset statistics

We chose fact verification as the target task to test our hypothesis for two reasons. First, in real-world fact-checking scenarios (Miranda et al., 2019; Nakov et al., 2021), retrieving evidence about niche or novel knowledge is crucial. If QE is effective only when the relevant knowledge has been seen during language model pretraining, its practical usefulness could be limited. Second, verdict prediction is a classification task, thus facil-

[1]AVeriTeC provides four categories, whereas FEVER and SciFact use three, excluding *conflicting evidence*. In SciFact, *CONTRADICT* is treated equivalently to *refuted*.

Method	FEVER			SciFact			AVeriTeC		
	Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
BM25	31	25	54	51.2	45.5	48.6	17.8	11.6	32
Performance by varying LLMs									
GPT-4o-mini	36.4±0.1	29.3±0.1	55.6±0.1	55.1±0.2	47.9±0.1	**52.5±0.5**	19.1±0.0	12.4±0.0	32.6±0.1
Claude-3-haiku	35.2±0.1	28.3±0.1	55.4±0.1	56±0.1	48.2±0.1	**52±0.5**	19.3±0.0	12.5±0.0	33.1±0.1
Gemini-1.5-flash	36.2±0.1	29.2±0.1	55.8±0.1	**56.2±0.2**	**49.4±0.1**	**52.2±0.5**	18.9±0.0	12.5±0.0	**33.3±0.2**
Llama-3.1-8b-it	35.7±0.1	28.6±0.2	55.6±0.2	54.9±0.2	47.8±0.2	51.9±0.3	19±0.0	12.4±0.0	32.2±0.2
Llama-3.1-70b-it	**38.3±0.1**	**31±0.1**	**56.1±0.1**	**56.4±0.3**	49.2±0.1	**52.4±0.7**	**19.3±0.1**	**12.7±0.0**	**33.4±0.2**
Mistral-7b-it	35.1±0.3	28±0.2	55.4±0.2	55.1±0.1	47.9±0.1	**51.9±0.6**	19.2±0.0	12.6±0.0	32.8±0.1
Mixtral-8x7b-it	35.1±0.2	27.9±0.2	55.3±0.2	54.6±0.2	47.7±0.1	51.9±0.4	**19.4±0.0**	**12.7±0.0**	33.2±0.1

(a) Query2doc

Method	FEVER			SciFact			AVeriTeC		
	Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Contriever	26.8	20.2	53.1	55.1	47.3	51.2	17.6	12.6	33.9
Performance by varying LLMs									
GPT-4o-mini	37.3±0.1	28.8±0.0	55.6±0.1	61.2±0.2	53.1±0.1	**54±0.5**	18.7±0.0	13.2±0.0	**35.7±0.6**
Claude-3-haiku	36.7±0.1	28.1±0.0	55.6±0.1	**62.8±0.1**	**54.7±0.1**	**53.7±0.4**	**19.3±0.0**	**14±0.0**	**36.2±0.6**
Gemini-1.5-flash	35±0.1	26.7±0.1	55.2±0.2	61±0.2	52.9±0.2	**53.5±0.7**	18±0.0	12.4±0.0	**35.7±0.5**
Llama-3.1-8b-it	36.7±0.1	28.4±0.1	55.4±0.1	61.2±0.2	53.4±0.2	**53.6±0.7**	18.9±0.0	13.6±0.0	35.5±0.4
Llama-3.1-70b-it	**40.4±0.2**	**31.7±0.2**	**55.9±0.1**	61.9±0.3	54.1±0.2	**53.6±0.5**	19±0.2	13.7±0.1	35.4±0.7
Mistral-7b-it	36.3±0.1	27.8±0.0	55.3±0.1	60.7±0.2	52.7±0.2	53.4±0.4	19±0.0	13.6±0.0	**35.8±0.7**
Mixtral-8x7b-it	37.6±0.1	29.1±0.1	55.7±0.1	61.3±0.2	53.1±0.1	53.3±0.3	19.2±0.0	13.7±0.0	**35.8±0.7**

(b) HyDE

Table 2: Fact verification performance using baseline retrievers and LLM-based query expansion methods, with the number of retrieved evidence set to five ($k = 5$). We report the average performance of query expansion methods along with standard errors, obtained by repeating the generations eight times.

itating clearer evaluation of how QE influences final outcomes compared to generation-based tasks, such as factual QA (Joshi et al., 2017; Kwiatkowski et al., 2019).

We employ three datasets that provide annotated evidence and corresponding veracity labels, along with an external knowledge store: FEVER (Thorne et al., 2018), SciFact (Wadden et al., 2020), and AVeriTeC (Schlichtkrull et al., 2023). While FEVER and SciFact contain verbatim extractions from K as gold evidence (i.e., E), AVeriTeC uses human-written evidence referencing K to verify claims. The three datasets cover diverse knowledge domains, as summarized in Table 1.

3.2 LLM-based Query Expansion

We evaluate two representative LLM-based QE methods, which expand queries by prompting an LLM.

Query2doc (Wang et al., 2023) generates a pseudo-document d based on a query q. It then forms an expanded query q^+ by concatenating d with multiple copies of q (Equation 1).

$$q^+ = concat(q \times n, d) \tag{1}$$

The expanded query q^+ is then used to retrieve documents via BM25 (Lin et al., 2021). Following Jagerman et al. (2023), we set n as 5.

HyDE (Gao et al., 2023) employs an LLM to generate hypothetical documents $[d_1, ..., d_N]$ to answer a query q. A dense retriever $g(\cdot)$ encodes q and each d_k separately, and their encoded embeddings are averaged to form the query vector v_{q^+} (Equation 2).

$$v_{q^+} = \frac{1}{N+1} \sum_{k=1}^{N} [g(d_k) + g(q)] \tag{2}$$

Here, we set N to 1 in this study. We use Contriever (Izacard et al., 2021) as $g(\cdot)$ with prompts provided in Appendix E.

3.3 Matching Method

Our goal is to determine whether an LLM-generated document d for a claim c contains a sentence entailed by the gold evidence $E = \{e_1, \cdots, e_m\}$. If such a sentence exists, it may indicate that the backbone LLM has already been exposed to data instances that cover the corresponding evidence. We employ a matching algorithm based on natural language inference (NLI), assigning each claim c to one of two conditions: *matched* (M) or *unmatched* ($\neg M$). The process has three steps. **(1) Sentence Segmentation:** Segment d into

Method	FEVER		SciFact		AVeriTeC	
	Query2doc	HyDE	Query2doc	HyDE	Query2doc	HyDE
GPT-4o-mini	75.8±0.1	**83.5±0.1**	52.8±0.4	59.1±0.7	40.4±0.2	68±0.3
Claude-3-haiku	76.6±0.1	77.8±0.1	56.1±0.4	53.8±0.2	40.8±0.1	62.3±0.2
Gemini-1.5-flash	69.9±0.3	70.2±0.3	50.8±0.6	**27.6±0.7**	44.1±0.1	59.6±0.3
Llama-3.1-8b-it	68.5±1.0	73±0.9	53.9±0.5	48.2±0.6	37.4±1.1	53.8±1.0
Llama-3.1-70b-it	78.3±0.7	71.7±0.7	57.5±0.3	55±0.8	48.1±0.8	47±4.8
Mistral-7b-it	72.5±0.2	75±0.2	51.1±0.5	49.4±0.8	44.7±0.2	55.6±0.3
Mixtral-8x7b-it	78.7±0.1	81±0.1	55.9±0.6	54.7±0.7	49.4±0.3	56.7±1.3

Table 3: Proportion of expanded queries containing sentences entailed by ground truth evidence across different LLM backbones.

sentences and remove reproductions of c to construct $S = \{s_1, \cdots, s_n\}$. **(2) NLI Labeling**: Use an NLI model to predict a label $l_{(i,j)}$ for each pair $(e_i, s_j) \in E \times S$, where $l_{(i,j)} \in \{$*entailment*, *contradiction*, *neutral*$\}$. **(3) Label Aggregation:** Aggregate all labels $\{l_{(1,1)}, \cdots, l_{(i,j)}, \cdots, l_{(m,n)}\}$ into a single label l. If at least one pair is labeled as *entailment*, assign *matched*, otherwise assigns *unmatched*.

We use the sentence segmentation module provided by spaCy[2], and employ GPT-4o-mini for NLI (Figure A3). To filter out claim reproductions, we apply ROUGE-2 (Lin, 2004) with a threshold of 0.95, based on manual inspection.

4 Experimental Results

We conducted evaluation experiments on three fact verification benchmarks. Each LLM-based generation was repeated eight times, and we report the average performance with the standard error.

Are LLM-based query expansion methods effective for fact verification? To assess the effectiveness of Query2doc and HyDE, we compared their performance against BM25 and Contriever that use c as query, respectively, as baseline retrievers. For evidence retrieval, we used Recall@k and NDCG@k ($k = 5$) as evaluation metrics (Manning, 2009) on the FEVER and SciFact datasets, where both the ground-truth evidence E and retrieved evidence $\tilde{E}$ come from the knowledge store K. In contrast, $\tilde{E}$ in AVeriTeC consists of human-written evidence rather than extracts from K. Therefore, following previous studies (Schlichtkrull et al., 2023; Chen et al., 2022), we applied the Hungarian algorithm (Kuhn, 1955) with METEOR (Banerjee and Lavie, 2005) and BERTScore (Zhang et al., 2020) on the top five retrieved sentences, computing token-level and embedding-level similarity, respectively, based on a binary assignment between generated and reference sequences. For verdict prediction, we used GPT-4o-mini with the five retrieved evidence and evaluated performance using macro F1. Evaluation details are provided in Appendix B.

As shown in Table 2, both Query2doc and HyDE consistently outperformed BM25 and Contriever across all three datasets and for seven different backbone LLMs (three proprietary and four open models). The performance gap between each baseline and its respective expansion method was statistically significant (p<0.001), demonstrating the effectiveness of the QE methods for evidence retrieval and, consequently, verdict prediction in these benchmarks. Results for $k = 10$ are available in Appendix F, showing a consistent trend.

Do LLM-generated documents include ground truth evidence? Table 3 presents the proportion of matched claims across the three datasets and seven LLMs. In most cases, more than 40% of the claims were matched, with a few exceptions. The lowest proportion (27.6%) was observed for SciFact when claims were expanded using HyDE with Gemini-1.5-flash—still a notable fraction. The highest proportion (83.5%) was observed for FEVER when using HyDE with GPT-4o-mini. Several examples of LLM-generated documents and gold evidence are provided in Table A5.

How does performance vary with the matching condition? Table 4 presents fact-verification performance based on whether LLM-generated documents contained sentences entailed by gold evidence, focusing on GPT-4o-mini. We observed a consistent trend across the three datasets and two expansion methods: *matched* claims (where LLM-generated documents contain sentences entailed

[2]https://huggingface.co/spacy/en_core_web_lg

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
BM25	ALL	31	25	54	51.2	45.5	48.6	17.8	11.6	32
Query2doc	ALL	36.4±0.1	29.3±0.1	55.6±0.1	55.1±0.2	47.9±0.1	52.5±0.5	19.1±0.0	12.4±0.0	32.6±0.1
	M	**40.5±0.1**	**32.8±0.1**	**58.4±0.0**	**63.3±0.4**	**57.1±0.3**	**53.7±0.3**	**21.6±0.1**	**17.6±0.1**	**38.3±0.3**
	$\neg M$	23.8±0.3	18.5±0.2	44.9±0.1	45.9±0.4	37.6±0.3	49±0.4	17.4±0.0	9±0.0	27.6±0.1
Contriever	ALL	26.8	20.2	53.1	55.1	47.3	51.2	17.6	12.6	33.9
HyDE	ALL	37.3±0.1	28.8±0.0	55.6±0.1	61.2±0.2	53.1±0.1	54±0.5	18.7±0.0	13.2±0.0	35.7±0.6
	M	**40±0.1**	**30.9±0.1**	**57.8±0.0**	**68.4±0.3**	**61.4±0.3**	**57.1±0.2**	**19.8±0.0**	**15.5±0.0**	**37±0.2**
	$\neg M$	23.4±0.4	17.9±0.3	41.9±0.2	50.8±0.5	41.2±0.4	48.9±0.4	16.4±0.0	8.3±0.1	30.3±0.4

Table 4: Fact verification performance based on whether documents generated by query expansion methods with GPT-4o-mini contain sentences entailed by gold evidence. Results for other LLMs are presented in Table A2.

by gold evidence) achieved significantly better performance than both all and *unmatched* claims, with statistical significance at p<0.001. Moreover, with a few exceptions, performance on *unmatched* claims was lower than that of the respective baseline methods without query expansion—BM25 for Query2doc and Contriever for HyDE. This trend was consistent across seven different LLMs and for $k = 10$, as shown in Table A2 and Table A3.

5 Discussion

Query2doc and HyDE boosted fact-verification performance across three benchmarks. Our findings provide empirical support for the effectiveness of LLM-based query expansion in zero-shot retrieval tasks (Gao et al., 2023; Jagerman et al., 2023; Lei et al., 2024; Wang et al., 2024a), particularly on fact verification benchmarks. They also highlights the crucial role of retrieving suitable evidence for claim verification (Chen et al., 2024a; Guo et al., 2022), suggesting a need for further research into evidence retrieval techniques, such as claim decomposition (Chen et al., 2022; Fan et al., 2020).

Documents generated by LLM-based query expansion methods frequently included sentences that were entailed by ground-truth evidence, indicating potential knowledge leakage. By applying an NLI-based matching algorithm, we examined whether LLMs reproduced gold evidence in response to query expansion prompts. Our results suggest that the seven LLMs studied in this paper were likely exposed to knowledge sources from the three benchmarks during training. This observation aligns with prior research on data leakage (Kandpal et al., 2023; Samuel et al., 2025) and memorization in LLMs (Cheng et al., 2024; Burns et al., 2022), representing the first empirical demonstration in the context of fact verification and query expansion.

Performance improvements from query expansion were consistent only when LLM-generated documents contained sentences entailed by gold evidence. This finding suggests that the success of hypothetical document generation, observed in Table 2 and prior studies (Gao et al., 2023; Wang et al., 2023; Yoon et al., 2024), may be largely attributable to LLM's internal knowledge encompassing benchmark knowledge sources. Furthermore, these results suggest that LLM-based query expansion may be limited in real-world scenarios requiring the retrieval of niche or novel knowledge, such as fact verification in the wild. Future advances in query expansion could incorporate external knowledge sources to address these limitations, as demonstrated in recent work (Lei et al., 2024).

6 Conclusion

This study examined the impact of two widely used LLM-based query expansion methods on fact verification. By applying NLI to the LLM-generated documents, we identified a consistent trend suggesting that knowledge leakage may be present in these benchmarks, inflating the perceived performance of these methods, particularly in real-world scenarios involving niche or novel knowledge. These findings highlight the need for future research on LLM-based query expansion methods that can effectively handle *unknown* queries, as well as the development of evaluation frameworks that more accurately reflect real-world settings.

Limitations

This study analyzed LLM *behaviors* to identify potential indicators of memorization and their impact on performance but did not investigate whether training on specific knowledge directly led to the generation of corresponding knowledge. Therefore, we cannot establish a causal relationship between data leakage and generation in response to query expansion prompts. Future research could further validate our methodology and findings by conducting experiments with synthetic data (Liu et al., 2024) or evaluating LLMs on genuinely novel knowledge (Kasai et al., 2024). To support the validity of the NLI-based automatic evaluation, we conducted a manual validation on sampled data and observed a consistent trend (Table A4) with that of the automatic method (Table 4).

Ethics and Impact Statement

We applied an NLI-based algorithm to investigate whether documents generated by LLM-based query expansion methods share underlying knowledge with gold evidence in fact verification benchmarks. Our findings suggest that knowledge leakage may exist; namely, LLMs may have been exposed to information related to the benchmarks' gold evidence during training and subsequently reproduced it in response to query-expansion prompts. This observation has important implications for both fact verification and broader benchmark-oriented NLP research, as it suggests that benchmark performance may be artificially inflated. Consequently, more trustworthy evaluation frameworks are needed to accurately reflect real-world scenarios.

Despite these observations, some caution is warranted when interpreting these findings. First, this study did not provide definitive evidence of knowledge leakage. As outlined in the Limitations, our research design cannot examine the causal relationship between model training data and the generated outputs. Future research could address this gap by conducting (continued) pretraining experiments using synthetic data or genuinely novel knowledge. Second, because our analysis focuses on three datasets within the fact-verification domain, the findings are limited in scope. Further experiments are necessary to determine whether these findings generalize to other tasks, such as factual QA (Joshi et al., 2017; Kwiatkowski et al., 2019). While the methodology presented here could be adapted to those settings, there remains a risk that the NLI model itself may introduce inaccuracies or biases toward particular labels. We conducted manual validation of the NLI results to mitigate these risks. Finally, we used ChatGPT to proofread portions of this manuscript.

Acknowledgments

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Graduate School of Metaverse Convergence support (IITP-2025-RS-2024-00430997) and Innovative Human Resource Development for Local Intellectualization (IITP-2025-RS-2022-00156360) programs, supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation). Kunwoo Park is the corresponding author.

References

Gianni Amati and Cornelis Joost Van Rijsbergen. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4):357–389.

Hiteshwar Kumar Azad and Akshay Deepak. 2019. Query expansion techniques for information retrieval: a survey. *Information Processing & Management*, 56(5):1698–1735.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*.

Jifan Chen, Grace Kim, Aniruddh Sriram, Greg Durrett, and Eunsol Choi. 2024a. Complex claim verification with evidence retrieved in the wild. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies (Volume 1: Long Papers), pages 3569–3587, Mexico City, Mexico. Association for Computational Linguistics.

Jifan Chen, Aniruddh Sriram, Eunsol Choi, and Greg Durrett. 2022. Generating literal and implied subquestions to fact-check complex claims. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3495–3516, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xinran Chen, Xuanang Chen, Ben He, Tengfei Wen, and Le Sun. 2024b. Analyze, generate and refine: Query expansion with LLMs for zero-shot open-domain QA. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11908–11922, Bangkok, Thailand. Association for Computational Linguistics.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Zhengfu He, Kai Chen, and Xipeng Qiu. 2024. Can AI assistants know what they don't know? In *Forty-first International Conference on Machine Learning*.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2023. Benchmark probing: Investigating data leakage in large language models. In *NeurIPS 2023 Workshop on Backdoors in Deep Learning-The Good, the Bad, and the Ugly*.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2024. Investigating data contamination in modern benchmarks for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8706–8719, Mexico City, Mexico. Association for Computational Linguistics.

Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Zhifang Sui, and Lei Li. 2024. Statistical knowledge assessment for large language models. *Advances in Neural Information Processing Systems*, 36.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR.

Angela Fan, Aleksandra Piktus, Fabio Petroni, Guillaume Wenzek, Marzieh Saeidi, Andreas Vlachos, Antoine Bordes, and Sebastian Riedel. 2020. Generating fact checking briefs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7147–7161, Online. Association for Computational Linguistics.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777, Toronto, Canada. Association for Computational Linguistics.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, William Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah Smith, and Hannaneh Hajishirzi. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. Association for Computational Linguistics.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query expansion by prompting large language models. *arXiv preprint arXiv:2305.03653*.

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

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A Smith, Yejin Choi, Kentaro Inui, et al. 2024. Realtime qa: what's the answer right now? *Advances in Neural Information Processing Systems*, 36.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

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*, 7:453–466.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, page 120–127, New York, NY, USA. Association for Computing Machinery.

Yibin Lei, Yu Cao, Tianyi Zhou, Tao Shen, and Andrew Yates. 2024. Corpus-steered query expansion with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 393–401.

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

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

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 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*, pages 2356–2362.

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*, 12:157–173.

Bill MacCartney. 2009. *Natural language inference*. Stanford University.

Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023a. Generative relevance feedback with large language models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2026–2031, New York, NY, USA. Association for Computing Machinery.

Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023b. Generative relevance feedback with large language models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2026–2031.

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

Christopher D Manning. 2009. *An introduction to information retrieval*.

Sebastião Miranda, David Nogueira, Afonso Mendes, Andreas Vlachos, Andrew Secker, Rebecca Garrett, Jeff Mitchel, and Zita Marinho. 2019. Automated fact checking in the news room. In *The world wide web conference*, pages 3579–3583.

Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, Giovanni Da San Martino, et al. 2021. Automated fact-checking for assisting human fact-checkers. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 4551–4558. International Joint Conferences on Artificial Intelligence Organization.

JJ Rocchio. 1971. Relevance feedback in information retrieval. *The SMART Retrieval System-Experiments in Automatic Document Processing/Prentice Hall.*

Vinay Samuel, Yue Zhou, and Henry Peng Zou. 2025. Towards data contamination detection for modern large language models: Limitations, inconsistencies, and oracle challenges. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5058–5070, Abu Dhabi, UAE. Association for Computational Linguistics.

Michael Schlichtkrull, Zhijiang Guo, and Andreas Vlachos. 2023. Averitec: a dataset for real-world claim verification with evidence from the web. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 65128–65167.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

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)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An extremely fast python interface to trec_eval. In *SIGIR*. ACM.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423, Singapore. Association for Computational Linguistics.

Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, Ruicheng Yin, Changze Lv, Xiaoqing Zheng, and Xuanjing Huang. 2024a. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17716–17736, Miami, Florida, USA. Association for Computational Linguistics.

Yuxia Wang, Minghan Wang, and Preslav Nakov. 2024b. Rethinking STS and NLI in large language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 965–982, St. Julian's, Malta. Association for Computational Linguistics.

Cheng Xu, Shuhao Guan, Derek Greene, M Kechadi, et al. 2024a. Benchmark data contamination of large language models: A survey. *arXiv preprint arXiv:2406.04244*.

Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. 2024b. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada. Association for Computational Linguistics.

Yejun Yoon, Jaeyoon Jung, Seunghyun Yoon, and Kunwoo Park. 2024. HerO at AVeriTeC: The herd of open large language models for verifying real-world claims. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 130–136, Miami, Florida, USA. Association for Computational Linguistics.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. R-tuning: Instructing large language models to say 'I don't know'. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7113–7139, Mexico City, Mexico. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2024. Dense text retrieval based on pretrained language models: A survey. *ACM Trans. Inf. Syst.*, 42(4).

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.

Appendix

A Target Dataset

This study used three fact verification benchmarks. From the BEIR benchmark (Thakur et al., 2021), we used the test set for FEVER and the train (505) and test (188) sets for SciFact. For AVeriTeC, we used the train (3,063) and dev (500) sets, as its test set is not publicly available. We excluded claims for which gold evidence was unavailable. Table 1 presents the descriptive statistics of the three datasets used in our experiments.

B Evaluation Metrics

Below, we describe the details of evaluation metric for evidence retrieval. Recall@K and NDCG@K are widely used evaluation metrics for information retrieval (Manning, 2009), adopted in this study for evaluating evidence retrieval performance for FEVER and SciFact. Recall@K assesses the proportion of relevant items retrieved within the top K results. NDCG@K measures the quality of ranked results by considering both the relevance of retrieved documents and their positions within the ranking. We used pyrec_eval (Van Gysel and de Rijke, 2018) to measure Recall@K and NDCG@K.

$$S(\hat{Y}, Y) = \frac{1}{|Y|} \max \sum_{\hat{y} \in \hat{Y}} \sum_{y \in Y} f(\hat{y}, y) X(\hat{y}, y) \tag{3}$$

For AVeriTeC, where gold evidence is not selected from a knowledge store but written by human annotators, we used METEOR and BERTScore with the Hungarian algorithm. Equation 3 presents the algorithm, where f is a pairwise scoring function, and X is a boolean function representing the assignment between the generated sequences $\hat{Y}$ and the reference sequences Y. To measure token-level and embedding-level similarity, we used METEOR and BERTScore for f, respectively. METEOR was computed using NLTK (Bird et al., 2009), and BERTScore was calculated with DeBERTa-xlarge-MNLI[3]. We reported observations for varying k: Table 2 and Table 4 for $k = 5$, and Table A1 and Table A3 for $k = 10$.

C Experimental Setups

For HyDE, we encoded queries and documents using Contriever[4]. For Query2doc, we used BM25 provided in PySerini (Lin et al., 2021). Following the same settings in previous study (Gao et al., 2023), we set LLM hyperparameters as follows: temperature as 0.7, top_p as 1.0, and max_tokens as 512. We used the Mann–Whitney U test for statistical testing on performance differences.

D Computing Environment

We ran experiments using two machines. The first machine has four Nvidia RTX A6000 GPUs (48GB per GPU) and 256GB RAM. The second machine has two Nvidia H100 GPUs (80GB per GPU) and 480 RAM. The experiments were conducted in a computing environment with the following configuration: Python 3.11.10, PyTorch 2.3.1, Transformers 4.43.4, vLLM 0.5.3, pyrec-eval 0.5, Faiss 1.8, Pyserini 0.40.0, NLTK 3.9.1, bert-score 0.3.13, rouge-score 0.1.2.

We used GPT-4o-mini, Claude 3 Haiku, Gemini 1.5 Flash via API, while Llama 3.1 (8B and 70B) and Mistral 7B, and Mixtral 8x7B were accessed through pretrained checkpoints. The model ids and parameter sizes are provided below.

- GPT-4o-mini: `gpt-4o-mini-2024-07-18` (Parameter size: unknown)
- Claude-3-haiku: `claude-3-haiku-20240307` (Parameter size: unknown)
- Gemini-1.5-flash: `gemini-1.5-flash` (Parameter size: unknown)
- Llama-3.1-8b-it: https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct (Parameter size: 8B)
- Llama-3.1-70b-it: https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct (Parameter size: 70B)
- Mistral-7b-it: https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3 (Parameter size: 7B)
- Mixtral-8x7b-it: https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1 (Parameter size: 46.7B)

E Prompts

Query Expansion Figure A1a, A1b, and A1c illustrate the HyDE prompts used in this study where the original prompt is adapted to each dataset. For Query2doc, we used the same prompt by following the suggestion in Wang et al. (2023), as shown in Figure A1d.

Verdict Prediction Figure A2 presents the prompt used for verdict prediction with GPT-4o-mini.

Natural Language Inference We used GPT-4o-mini for natural language inference, employing a prompt proposed in a previous study (Wang et al., 2024b), as illustrated in Figure A3. To support its validity, two authors manually annotated the labels

[3] https://huggingface.co/microsoft/deberta-xlarge-mnli

[4] https://huggingface.co/facebook/contriever

Please write a wikipedia passage to verify the claim.
Claim: [CLAIM]
Passage: **[OUTPUT]**

(a) The prompt used for HyDE in the FEVER dataset.

Please write a scientific paper passage to support/refute the claim.
Claim: [CLAIM]
Passage: **[OUTPUT]**

(b) The prompt used for HyDE in the SciFact dataset.

Please write a fact-checking article to verify the claim.
Claim: [CLAIM]
Passage: **[OUTPUT]**

(c) The prompt used for HyDE in the AVeriTeC dataset.

Write a passage that answers the following query:
[CLAIM]
[OUTPUT]

(d) The prompt used for Query2doc.

Figure A1: Prompts used for query expansion.

for a randomly selected set of 200 pairs following the guidelines presented in Figure A4. The GPT-based NLI model achieved an F1 score of 0.8 on the sampled data.

F Supplementary Results

LLM comparison for fact verification Table 2 present the results for evidence retrieval and verdict prediction by varying backbone LLMs for query expansion. We observed that Llama-3.1-70b-it generally performed well when used with Query2doc. For HyDE, while Llama-3.1-70b-it achieved the best performance on FEVER, Claude-3-haiku obtained higher evaluation scores in SciFact and AVeriTeC.

Proportion of matched claims across different benchmarks Table 3 presents the distribution of matched claims across three datasets, varying the LLMs used for query expansion. On average, the estimated proportion was higher for FEVER than for the other two datasets. While FEVER was constructed using public Wikipedia documents, SciFact is based on scientific literature, covering *niche* knowledge, and AVeriTeC is the most recent dataset based on web documents collected by human annotators, covering *recent* knowledge. Given these characteristics, the highest proportion observed in FEVER partially supports the validity of the NLI-based estimation.

Your task is to predict the verdict of a claim based on the provided evidence. Select one of the following labels: [LABEL]. Generate only the label without additional explanation or content.

Claim: [CLAIM]

Evidence 1: [EVIDENCE 1]
...
Evidence N: [EVIDENCE N]

Label: **[OUTPUT]**

Figure A2: The prompt used for fact verification with GPT-4o-mini. N denotes the total number of retrieved evidence.

Given the premise sentence S1, determine if the hypothesis sentence S2 is entailed or contradicted or neutral, by three labels: entailment, contradiction, neutral.
Respond only with one of the labels.
S1: [GOLD EVIDENCE]
S2: [LLM-GENERATED SENTENCE]
Label: **[OUTPUT]**

Figure A3: The prompt used for NLI.

Effects of retrieving more evidence Table A1 presents the fact verification performance with an increased number of retrieved evidence ($k = 10$). Performance improves in every case across different LLMs and datasets compared to the results with $k = 5$. Table A3 shows performance depending on the matching condition, showing a consistent trend with Table A2 when retrieving more evidence.

Manual annotation To support the validity of the NLI-based matching algorithm, we conducted manual annotations on a sampled dataset. Two authors independently reviewed documents generated by Query2doc and HyDE for all 500 samples in the AVeriTeC development set, following the guideline presented in Figure A5. A claim was labeled as *matched* if the LLM-generated document contained all or part of any gold evidence. For the backbone LLMs, we used Llama-3.1-70b-it for Query2doc and Claude-3-haiku for HyDE, as these models achieved the best performance for their respective methods. The two annotators achieved

Method	FEVER			SciFact			AVeriTeC		
	Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
BM25	37.1	27.1	55.3	58.4	48.3	50.6	21	15	33.1
Performance by varying LLMs									
GPT-4o-mini	44.2±0.1	32±0.1	57±0.1	64±0.0	51.4±0.1	**53.2±0.6**	22.3±0.0	15.9±0.0	34.6±0.1
Claude-3-haiku	43.4±0.1	31±0.1	56.9±0.2	64.3±0.2	51.5±0.1	**53±0.5**	22.5±0.0	16±0.0	34.7±0.1
Gemini-1.5-flash	43.4±0.1	31.7±0.0	56.9±0.1	**64.7±0.2**	**52.8±0.1**	**53.1±0.6**	22.3±0.0	16.2±0.0	**34.7±0.2**
Llama-3.1-8b-it	43.6±0.1	31.3±0.1	56.8±0.1	63.2±0.3	51.2±0.1	**53±0.5**	22.3±0.0	15.9±0.0	34.5±0.1
Llama-3.1-70b-it	**46.1±0.2**	**33.7±0.1**	**57.2±0.2**	**64.6±0.2**	52.5±0.1	**53.1±0.3**	**22.7±0.1**	**16.3±0.0**	**34.8±0.1**
Mistral-7b-it	43.2±0.2	30.8±0.2	56.8±0.1	63.1±0.2	51±0.1	52.6±0.4	22.5±0.0	16.2±0.0	34.6±0.1
Mixtral-8x7b-it	43.4±0.2	30.8±0.2	56.8±0.2	63.6±0.2	51.3±0.1	**52.8±0.4**	22.6±0.0	16.1±0.0	**34.9±0.1**

(a) Query2doc

Method	FEVER			SciFact			AVeriTeC		
	Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Contriever	34.4	22.8	54.8	65	51.2	54	20.8	16.1	34.8
Performance by varying LLMs									
GPT-4o-mini	46.7±0.1	32.1±0.0	56.7±0.1	70±0.1	56.7±0.1	54.6±0.3	22.3±0.0	17±0.0	36.6±0.4
Claude-3-haiku	46.2±0.0	31.4±0.0	56.7±0.1	**71.6±0.2**	**58.3±0.1**	**55±0.3**	**22.8±0.0**	**17.8±0.0**	**37.6±0.3**
Gemini-1.5-flash	44.2±0.1	29.9±0.1	56.5±0.1	69.8±0.1	56.6±0.2	**54.8±0.3**	21.6±0.0	16.3±0.0	**37.1±0.8**
Llama-3.1-8b-it	46.4±0.1	31.8±0.1	56.6±0.1	70.1±0.2	57±0.1	54.3±0.5	22.4±0.0	17.4±0.0	36.8±0.3
Llama-3.1-70b-it	**49.7±0.2**	**35±0.2**	**57±0.1**	70.8±0.2	57.8±0.2	**55±0.5**	22.5±0.2	17.5±0.1	**36.7±0.9**
Mistral-7b-it	46.1±0.1	31.2±0.0	56.5±0.1	69.8±0.2	56.4±0.2	**54.8±0.3**	22.7±0.0	17.5±0.0	**37.3±0.5**
Mixtral-8x7b-it	47.6±0.0	32.6±0.0	56.9±0.2	70.2±0.2	56.8±0.1	**54.8±0.5**	**22.8±0.0**	17.6±0.0	**37.2±0.5**

(b) HyDE

Table A1: Fact verification performance using baseline retrievers and LLM-based query expansion methods, with the number of retrieved evidence set to ten ($k = 10$). We report the average performance of query expansion methods along with standard errors, obtained by repeating the generations eight times.

Premise: [GOLD EVIDENCE]
Hypothesis:
[LLM-GENERATED SENTENCE]

Given the premise, determine whether the hypothesis is entailed.

☐ **Entailment**

☐ **Non-Entailment**

Figure A4: Manual labeling guidelines for natural language inference.

Claim: [CLAIM]
Gold Evidence: [GOLD EVIDENCE]
LLM-generated Document:
[LLM-GENERATED DOCUMENT]

Determine whether the LLM-generated document contains the whole or part of any gold evidence.

☐ **Included**

☐ **Not Included**

Figure A5: Manual labeling guidelines for determining whether LLM-generated documents contain gold evidence.

a high-level of inter-annotator agreement, with a Cohen's kappa of 0.837 across 1,000 generations. The estimated proportion of matched claims were 50.8% and 59.6%, respectively, closely aligning with those from the NLI-based method, with differences falling within the error margin.

Table A4 presents performance depending on manually annotated matching conditions, showing a consistent trend with the results from the NLI-based method (Table 4 and Table A2).

G Qualitative Analysis

Table A5 presents examples of LLM-generated documents along with gold evidence. In example (a), the claim concerns Nigeria's history, and the gold evidence specifies the period under military rule, which was reproduced in the generated document. Example (b) pertains to U.S. Supreme Court Justice Ruth Bader Ginsberg, where the gold evidence provides her bibliography and medical history. The LLM-generated text includes this information along with specific years. Notably, it also introduces an additional fact about lung cancer, which is not covered by the gold evidence. Examples (c) and (d) illustrate unmatched cases where the generated text contains factual errors.

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	35.2±0.1	28.3±0.1	55.4±0.1	56 ±0.1	48.2±0.1	52±0.5	19.3±0.0	12.5±0.0	33.1±0.1
	M	**38.7±0.1**	**31.2±0.1**	**58±0.0**	**63.7±0.3**	**56.7±0.3**	**54.1±0.2**	**22±0.0**	**17.7±0.0**	**38.8±0.3**
	$\neg M$	23.8±0.1	18.5±0.1	44.7±0.1	46.2±0.6	37.2±0.4	48±0.5	17.4±0.0	8.9±0.0	27.6±0.1
HyDE	ALL	36.7±0.1	28.1±0.0	55.6±0.1	62.8±0.1	54.7±0.1	53.7±0.4	19.3±0.0	14 ±0.0	36.2±0.6
	M	**39.9±0.2**	**30.7±0.1**	**57.5±0.1**	**71.1±0.5**	**63.8±0.4**	**57.5±0.3**	**20.4±0.1**	**16.5±0.1**	**37.8±0.3**
	$\neg M$	25.5±0.2	18.9±0.2	45±0.2	53±0.5	44.1±0.5	48.8±0.4	17.3±0.1	9.8±0.1	31.9±0.3

(a) Claude-3-haiku

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	36.2±0.1	29.2±0.1	55.8±0.1	56.2±0.2	49.4±0.1	52.2±0.5	18.9±0.0	12.5±0.0	33.3±0.2
	M	**38.5±0.1**	**31.4±0.0**	**58.8±0.0**	**63.4±0.5**	**56.9±0.4**	**55±0.5**	**20.5±0.0**	**16.5±0.1**	**38.1±0.4**
	$\neg M$	30.7±0.1	24.1±0.1	48±0.2	48.9±0.4	41.6±0.3	48.7±0.3	17.6±0.0	9.4±0.1	28.5±0.2
HyDE	ALL	35±0.1	26.7±0.1	55.2±0.2	61±0.2	52.9±0.2	53.5±0.7	18±0.0	12.4±0.0	35.7±0.5
	M	**37.1±0.1**	**28.2±0.1**	**59±0.1**	**67.1±0.6**	**60.1±0.4**	**57.3±0.5**	**18.8±0.0**	**14.7±0.1**	**37.3±0.2**
	$\neg M$	30.1±0.1	23.1±0.1	45±0.2	58.7±0.4	50.2±0.2	52±0.3	16.8±0.1	9.2±0.1	31.6±0.3

(b) Gemini-1.5-flash

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	35.7±0.1	28.6±0.2	55.6±0.2	54.9±0.2	47.8±0.2	51.9±0.3	19±0.0	12.4±0.0	32.2±0.2
	M	**39±0.2**	**31.4±0.2**	**58.6±0.1**	**63.9±0.4**	**57.5±0.4**	**55.1±0.4**	**21.7±0.1**	**17.3±0.1**	**36.6±0.3**
	$\neg M$	28.5±0.3	22.3±0.3	48.5±0.3	44.2±0.6	36.5±0.4	47.6±0.3	17.4±0.0	9.4±0.1	28.5±0.1
HyDE	ALL	36.7±0.1	28.4±0.1	55.4±0.1	61.2±0.2	53.4±0.2	53.6±0.7	18.9±0.0	13.6±0.0	35.5±0.4
	M	**39.5±0.1**	**30.6±0.1**	**58.2±0.1**	**68.7±0.7**	**62.3±0.6**	**56.3±0.4**	**20±0.1**	**16.1±0.1**	**37.7±0.2**
	$\neg M$	29.2±0.2	22.4±0.2	46.8±0.2	54.1±0.8	45.1±0.6	50.6±0.4	17.7±0.1	10.6±0.1	31.8±0.3

(c) Llama-3.1-8b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	38.3±0.1	31±0.1	56.1±0.1	56.4±0.3	49.2±0.1	52.4±0.7	19.3±0.1	12.7±0.0	33.4±0.2
	M	**41.3±0.1**	**33.6±0.1**	**58.6±0.1**	**65±0.3**	**58.3±0.2**	**54.9±0.5**	**21.6±0.1**	**17.2±0.1**	**38.1±0.3**
	$\neg M$	27.6±0.4	21.7±0.4	45.9±0.3	44.9±0.5	37±0.3	47.9±0.2	17.3±0.1	8.6±0.1	27.6±0.2
HyDE	ALL	40.4±0.2	31.7±0.2	55.9±0.1	61.9±0.3	54.1±0.2	53.6±0.5	19±0.2	13.7±0.1	35.4±0.7
	M	**44.3±0.5**	**35±0.5**	**58.4±0.1**	**69.2±0.4**	**62.1±0.4**	**56.7±0.3**	**20.9±0.2**	**17.3±0.3**	**38.3±0.2**
	$\neg M$	30.4±0.3	23.3±0.3	48.9±0.2	52.9±0.7	44.3±0.6	48.9±0.5	17.4±0.1	10.6±0.2	31.5±0.4

(d) Llama-3.1-70b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	35.1±0.3	28±0.2	55.4±0.2	55.1±0.1	47.9±0.1	51.9±0.6	19.2±0.0	12.6±0.0	32.8±0.1
	M	**39±0.2**	**31.3±0.2**	**58.2±0.1**	**64.8±0.3**	**58±0.2**	**54.8±0.5**	**21.5±0.1**	**17.3±0.1**	**37.5±0.5**
	$\neg M$	24.8±0.4	19.1±0.3	46.7±0.2	44.9±0.5	37.2±0.3	48.3±0.3	17.4±0.0	8.8±0.1	27.7±0.2
HyDE	ALL	36.3±0.1	27.8±0.0	55.3±0.1	60.7±0.2	52.7±0.2	53.4±0.4	19±0.0	13.6±0.0	35.8±0.7
	M	**39.2±0.2**	**30.1±0.1**	**57.7±0.1**	**67.8±0.5**	**60.9±0.4**	**55.5±0.7**	**20.1±0.1**	**16±0.1**	**36.8±0.4**
	$\neg M$	27.5±0.2	20.9±0.1	46.4±0.2	53.8±0.3	44.8±0.3	50.4±0.3	17.7±0.1	10.6±0.1	33.1±0.4

(e) Mistral-7b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@5	NDCG@5	F1	Recall@5	NDCG@5	F1	METEOR	BERTScore	F1
Query2doc	ALL	35.1±0.2	27.9±0.2	55.3±0.2	54.6±0.2	47.7±0.1	51.9±0.4	19.4±0.0	12.7±0.0	33.2±0.1
	M	**38.6±0.2**	**30.9±0.2**	**57.8±0.1**	**63.5±0.5**	**57.3±0.4**	**54.1±0.3**	**21.5±0.1**	**17±0.1**	**37.3±0.3**
	$\neg M$	22.3±0.3	17.1±0.2	44.3±0.2	43.4±0.4	35.6±0.3	47.8±0.3	17.3±0.1	8.4±0.1	27.7±0.3
HyDE	ALL	37.6±0.1	29.1±0.1	55.7±0.1	61.3±0.2	53.1±0.1	53.3±0.3	19.2±0.0	13.7±0.0	35.8±0.7
	M	**40.3±0.1**	**31.2±0.1**	**57.7±0.0**	**68.9±0.4**	**61.4±0.3**	**55.4±0.2**	**20.3±0.1**	**16.1±0.1**	**37.1±0.2**
	$\neg M$	26.4±0.2	20.4±0.2	45.2±0.2	52.2±0.3	43.1±0.2	49.7±0.4	17.7±0.1	10.6±0.1	32.6±0.4

(f) Mixtral-8x7b-it

Table A2: Fact verification performance depending on whether the document generated by query expansion methods contains sentences entailed by gold evidence, with the number of retrieved evidence set to five ($k = 5$). We report performance using different backbone LLMs for query expansion.

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	44.2±0.1	32±0.1	57±0.1	64±0.0	51.4±0.1	53.2±0.6	22.3±0.0	15.9±0.0	34.6±0.1
	M	**48.8±0.1**	**35.7±0.1**	**59.6±0.1**	**71.7±0.3**	**60.6±0.3**	**54.1±0.6**	**25.3±0.1**	**21.4±0.1**	**40.6±0.3**
	$\neg M$	29.9±0.2	20.5±0.1	47.3±0.2	55.4±0.4	41.2±0.3	50.1±0.4	20.3±0.0	12.2±0.0	29.2±0.2
HyDE	ALL	46.7±0.1	32.1±0.0	56.7±0.1	70±0.1	56.7±0.1	54.6±0.3	22.3±0.0	17±0.0	36.6±0.4
	M	**50.2±0.1**	**34.5±0.0**	**58.8±0.0**	**76.5±0.3**	**64.8±0.2**	**57.7±0.2**	**23.6±0.0**	**19.5±0.0**	**38.1±0.2**
	$\neg M$	29.4±0.4	20±0.3	44.2±0.2	60.5±0.4	44.9±0.4	49.5±0.3	19.4±0.1	11.8±0.1	31±0.1

(a) GPT-4o-mini

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	43.4±0.1	31±0.1	56.9±0.2	64.3±0.2	51.5±0.1	53±0.5	22.5±0.0	16±0.0	34.7±0.1
	M	**47.3±0.1**	**34.2±0.1**	**59.2±0.1**	**70.9±0.3**	**59.7±0.3**	**55.3±0.3**	**25.7±0.1**	**21.5±0.0**	**40.4±0.4**
	$\neg M$	30.5±0.2	20.7±0.1	47.3±0.2	55.8±0.6	40.9±0.4	49.3±0.2	20.4±0.1	12.2±0.1	29.3±0.2
HyDE	ALL	46.2±0.0	31.4±0.0	56.7±0.1	71.6±0.2	58.3±0.1	55±0.3	22.8±0.0	17.8±0.0	37.6±0.3
	M	**50±0.1**	**34.3±0.1**	**58.5±0.1**	**78.9±0.3**	**67.1±0.4**	**58.7±0.2**	**24.2±0.1**	**20.4±0.1**	**38.9±0.2**
	$\neg M$	32.8±0.3	21.4±0.2	46.6±0.2	63±0.6	48.1±0.6	50.5±0.3	20.5±0.1	13.5±0.1	33.6±0.3

(b) Claude-3-haiku

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	43.4±0.1	31.7±0.0	56.9±0.1	64.7±0.2	52.8±0.1	53.1±0.6	22.3±0.0	16.2±0.0	34.7±0.2
	M	**46.3±0.0**	**34.1±0.0**	**59.8±0.1**	**71.4±0.3**	**60.2±0.3**	**55.3±0.3**	**24.4±0.0**	**20.4±0.1**	**39.5±0.3**
	$\neg M$	36.7±0.1	26.1±0.1	49.7±0.1	57.9±0.4	45.1± 0.3	49.9±0.3	20.7±0.0	12.8±0.0	29.9±0.3
HyDE	ALL	44.2±0.1	29.9±0.1	56.5±0.1	69.8±0.1	56.6±0.2	54.8±0.3	21.6±0.0	16.3±0.0	37.1±0.8
	M	**47.1±0.1**	**31.8±0.1**	**59.9±0.0**	**75.5±0.5**	**63.8±0.4**	**58.3±0.3**	**22.6±0.0**	**18.6±0.1**	**38.5±0.3**
	$\neg M$	37.5±0.2	25.6±0.1	47.6±0.2	67.7±0.3	53.8±0.2	53.4±0.2	20.1±0.1	12.9±0.1	33.1±0.4

(c) Gemini-1.5-flash

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	43.6±0.1	31.3±0.1	56.8±0.1	63.2±0.3	51.2±0.1	53±0.5	22.3±0.0	15.9±0.0	34.5±0.1
	M	**47.2±0.1**	**34.3±0.2**	**59.5±0.0**	**71.2±0.3**	**60.6±0.4**	**55.7±0.4**	**25.4±0.1**	**21±0.1**	**38.5±0.1**
	$\neg M$	35.8±0.3	24.8±0.3	50.4±0.2	53.8±0.7	40.2±0.5	49.3±0.2	20.4±0.0	12.8±0.1	30.7±0.2
HyDE	ALL	46.4±0.1	31.8±0.1	56.6±0.1	70.1±0.2	57±0.1	54.3±0.5	22.4±0.0	17.4±0.0	36.8±0.3
	M	**50±0.1**	**34.3±0.1**	**59.1±0.1**	**77.7±0.6**	**66.1±0.4**	**57.2±0.3**	**23.7±0.1**	**19.9±0.1**	**38.9±0.2**
	$\neg M$	36.7±0.3	24.9±0.2	48.8±0.2	63±0.7	48.6±0.5	51.4±0.4	21±0.1	14.3±0.1	33.3±0.3

(d) Llama-3.1-8b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	46.1±0.2	33.7±0.1	57.2±0.2	64.6±0.2	52.5±0.1	53.1±0.3	22.7±0.1	16.3±0.0	34.8±0.1
	M	**49.4±0.2**	**36.4±0.1**	**59.5±0.0**	**72.5±0.4**	**61.4±0.2**	**55.4±0.4**	**25.4±0.1**	**21.1±0.1**	**39.6±0.3**
	$\neg M$	34.2±0.2	23.9±0.3	48.1±0.2	54±0.5	40.5±0.2	48.9±0.2	20.2±0.0	11.9±0.1	28.9±0.3
HyDE	ALL	49.7±0.2	35±0.2	57±0.1	70.8±0.2	57.8±0.2	55±0.5	22.5±0.2	17.5±0.1	36.7±0.9
	M	**54.4±0.4**	**38.6±0.4**	**59.3±0.0**	**77.5±0.5**	**65.6±0.4**	**57.8±0.3**	**24.7±0.2**	**21.3±0.3**	**39.6±0.3**
	$\neg M$	37.9±0.2	25.9±0.2	50.7±0.2	62.7±0.6	48.2±0.6	50.8±0.3	20.7±0.1	14.2±0.2	32.9±0.4

(e) Llama-3.1-70b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	43.2±0.2	30.8±0.2	56.8±0.1	63.1±0.2	51±0.1	52.6±0.4	22.5±0.0	16.2±0.0	34.6±0.1
	M	**47.5±0.2**	**34.3±0.2**	**59.2±0.0**	**71.5±0.4**	**60.8±0.3**	**55.1±0.2**	**25.2±0.1**	**21.1±0.1**	**39.8±0.3**
	$\neg M$	31.8±0.4	21.5±0.3	48.9±0.1	54.3±0.5	40.8±0.2	49.4±0.3	20.4±0.0	12.2±0.1	29.3±0.2
HyDE	ALL	46.1±0.1	31.2±0.0	56.5±0.1	69.8±0.2	56.4±0.2	54.8±0.3	22.7±0.0	17.5±0.0	37.3±0.5
	M	**50±0.2**	**33.9±0.1**	**58.9±0.1**	**76.7±0.4**	**64.7±0.4**	**57.4±0.7**	**23.9±0.1**	**20.1±0.1**	**38.5±0.4**
	$\neg M$	34.3±0.2	23.3±0.1	48.4±0.2	63±0.2	48.5±0.2	51.5±0.3	21.1±0.1	14.3±0.1	34.4±0.3

(f) Mistral-7b-it

Method	Data	FEVER			SciFact			AVeriTeC		
		Recall@10	NDCG@10	F1	Recall@10	NDCG@10	F1	METEOR	BERTScore	F1
Query2doc	ALL	43.4±0.2	30.8±0.2	56.8±0.2	63.6±0.2	51.3±0.1	52.8±0.4	22.6±0.0	16.1±0.0	34.9±0.1
	M	**47.3±0.2**	**33.9±0.2**	**59±0.1**	**71.3±0.5**	**60.6±0.4**	**54.6±0.6**	**25.1±0.1**	**20.7±0.1**	**39±0.2**
	$\neg M$	28.7±0.3	19.2±0.2	46.7±0.2	53.8±0.5	39.5±0.3	49.2±0.4	20.2±0.1	11.7±0.1	29.5±0.2
HyDE	ALL	47.6±0.0	32.6±0.0	56.9±0.2	70.2±0.2	56.8±0.1	54.8±0.5	22.8±0.0	17.6±0.0	37.2±0.5
	M	**50.9±0.1**	**34.9±0.0**	**58.8±0.0**	**77.7±0.1**	**65.2±0.2**	**56.8±0.3**	**24.2±0.0**	**20.1±0.1**	**38.7±0.3**
	$\neg M$	33.5±0.3	22.8±0.2	47.2±0.1	61±0.4	46.6±0.3	51.7±0.4	21±0.1	14.3±0.1	33.8±0.4

(g) Mixtral-8x7b-it

Table A3: Fact verification performance depending on whether the document generated by query expansion methods contains sentences entailed by gold evidence, with the number of retrieved evidence set to ten ($k = 10$). We report performance using different backbone LLMs for query expansion.

Method	Data	METEOR	BERTScore	F1
Query2doc	ALL	20.5	13.8	34.7
	M	**21.6**	**15.4**	**40.9**
	$\neg M$	19.4	12.2	27.3
HyDE	ALL	19.9	14.7	38.7
	M	**20.7**	**16.1**	**40.9**
	$\neg M$	18.8	12.5	33.6

(a) $k = 5$

Method	Data	METEOR	BERTScore	F1
Query2doc	ALL	23.6	17.3	36.7
	M	**24.9**	**19**	**45.2**
	$\neg M$	22.2	15.5	26.6
HyDE	ALL	23.9	18.2	38.3
	M	**24.6**	**19.7**	**39.4**
	$\neg M$	22.7	16.1	34.9

(b) $k = 10$

Table A4: Fact verification performance on 500 samples from the AVeriTeC development set using manually annotated NLI labels. For query expansion, we used Llama-3.1-70b-it for Query2doc and Claude-3-haiku for HyDE, as each model achieved the best performance for its respective expansion method. GPT-4o-mini was used for verdict prediction.

Claim	U.S. President Donald Trump and his wife, First Lady Melania Trump, have tested positive for COVID-19
Gold Evidence	• Donald trump tweeted that he had tested positive for Covid 19 on 1st October 2020 • Donald trump tweeted that he and first lady Melania had tested positive for Covid 19 on 1st October 2020 • The President's physician, Navy Cmdr. Dr. Sean Conley, wrote that he received confirmation of the positive tests on Thursday evening.
Generation	On October 1, 2020, it was announced that U.S. President Donald Trump and his wife, First Lady Melania Trump, had tested positive for COVID-19. This news came after one of the President's closest advisors, Hope Hicks, had also tested positive for the virus earlier that day. The President and the First Lady began experiencing symptoms shortly after Hicks' diagnosis and were subsequently tested. Both Trumps received their positive test results in the early hours of October 2, 2020. Following their diagnosis, the President and the First Lady immediately began self-isolating in the White House. President Trump's physician, Dr. Sean Conley, issued a statement saying that the President and the First Lady would remain in quarantine for the next 14 days, as per Centers for Disease Control and Prevention (CDC) guidelines. The news of the President's positive test result sparked widespread concern and reaction, with leaders from around the world offering their well wishes for a speedy recovery. The diagnosis also led to increased scrutiny of the White House's handling of the pandemic, particularly with regards to the safety protocols in place at the time. Fortunately, both President Trump and the First Lady made a full recovery from the virus. President Trump was hospitalized for a few days at Walter Reed National Military Medical Center, where he received treatment and was later discharged on October 5, 2020.

(a) Matched example

Claim	US Supreme Court Justice, Ruth Bader Ginsburg, fought cancer for decades before she passed away in September 2020.
Gold Evidence	• Ruth Bader Ginsburg was a U.S. Supreme Court justice, the second woman to be appointed to the position. • On September 18, 2020, Justice Ruth Bader Ginsburg, the second woman to serve on the Supreme Court of the United States, passed away at the age of eighty-seven • She died of complications from metastatic pancreatic cancer. • colon cancer (first diagnosed in 1999), pancreatic cancer (2009)
Generation	Ruth Bader Ginsburg, a trailblazing U.S. Supreme Court Justice, demonstrated unwavering resilience in the face of adversity, including her prolonged battle with cancer. Throughout her life, she faced numerous health challenges, including colorectal cancer in 1999, pancreatic cancer in 2009, and lung cancer in 2018. Furthermore, in 2019, she underwent radiation therapy for a tumor on her pancreas. Despite these setbacks, Ginsburg continued to serve on the Supreme Court, inspiring countless individuals with her strength, determination, and unwavering commitment to her work. Tragically, her courageous fight against cancer came to an end on September 18, 2020, when she passed away at the age of 87, leaving behind an enduring legacy as a champion of gender equality, social justice, and the rule of law.

(b) Matched example

Claim	While serving as Town Supervisor on Grand Island, Nebraska, US Nate McMurray voted to raise taxes on homeowners.
Gold Evidence	• Grand Island raised its tax levy on November 8th 2018. • A Public Hearing was held on Thursday, November 8, 2018 at 7:00p.m. for the purpose of hearing anyone who wants to comment on the proposed 2019 Town of Grand Island budget. Supervisor Nate McMurray closed the Public Hearing.
Generation	After researching Nate McMurray's political career, there is no evidence that he has ever served as Town Supervisor on Grand Island, Nebraska. Grand Island is a city in central Nebraska, not a town, and it does not have a position of Town Supervisor. Nate McMurray is a former town councilman in the town of Grand Island, New York, which is located in Erie County, New York, not Nebraska. However, there are no reports or records indicating that McMurray voted to raise taxes on homeowners while serving on the Grand Island, New York town council. In conclusion, the claim that Nate McMurray voted to raise taxes on homeowners while serving as Town Supervisor on Grand Island, Nebraska is false. Nate McMurray has never held a political position in the state of Nebraska, and the town of Grand Island, Nebraska does not exist. This appears to be a fabricated claim with no basis in fact.

(c) Unmatched example

Claim	U.S. mortalities this year are not materially different from the previous 5 years
Gold Evidence	• The cumulative number of deaths in 2020 is far greater than the cumulative number of deaths of any year from 2015 to 2019, as shown in the figures in link below. • The Historical data on death rate shows that there were more deaths recorded in the singular year 2020 compared to the death rate rate in each year from 2015-2019
Generation	**U.S. Mortality Rates Remain Steady, Showing No Significant Increase from Previous Years** According to recent data, mortality rates in the United States for this year are not showing any significant deviations from the past five years. Despite concerns about various health crises and trends, the overall number of deaths in the country has remained relatively stable. When analyzing the data, it becomes clear that the mortality rates for this year are largely in line with the averages seen in the previous five years. This suggests that, despite fluctuations in specific causes of death or regional variations, the overall trend in mortality rates has not undergone a significant shift. Several factors could be contributing to this stability, including advances in medical technology, improvements in healthcare access and quality, and public health initiatives aimed at reducing mortality rates. Additionally, the country's demographics, such as population size and age distribution, have not changed dramatically enough to impact mortality rates substantially. While specific causes of death, such as deaths from accidents, diseases, or other factors, may have seen some variation, the overall mortality rate has not been significantly affected. This steady trend is a positive sign, indicating that the country's healthcare system and public health efforts are effectively managing and mitigating mortality risks. It is essential to note, however, that mortality rates can fluctuate over time due to various factors, including emerging health crises, changes in population demographics, or shifts in healthcare policies. Therefore, continuous monitoring and analysis of mortality data are crucial to identifying potential trends and areas for improvement. In conclusion, the current mortality rates in the United States are not materially different from the previous five years, indicating a stable trend. This stability is a testament to the country's healthcare system and public health efforts, but it also highlights the need for continued vigilance and monitoring to address potential future challenges.

(d) Unmatched example

Table A5: Examples of generated documents and gold evidence for the target claims. Colored highlights indicate information in the generated documents that overlaps with the gold evidence.