GenGO Ultra: an LLM-powered ACL Paper Explorer

Sotaro Takeshita, Tornike Tsereteli, Simone Paolo Ponzetto
Data and Web Science Group, University of Mannheim, Germany
{sotaro.takeshita, tornike.tsereteli, ponzetto}@uni-mannheim.de

Abstract

The ever-growing number of papers in natural language processing (NLP) poses the challenge of finding relevant papers. In our previous paper, we introduced *GenGO* (Takeshita et al., 2024b), which complements NLP papers with various information, such as aspect-based summaries, to enable efficient paper exploration. While it delivers a better literature search experience, it lacks an interactive interface that dynamically produces information tailored to the user's needs. To this end, we present an extension to our previous system, dubbed *GenGO Ultra*, which exploits large language models (LLMs) to dynamically generate responses grounded by published papers. We also conduct multi-granularity experiments to evaluate six text encoders and five LLMs. Our system is designed for transparency – based only on open-weight models, visible system prompts, and an open-source code base – to foster further development and research on top of our system: `https://gengo-ultra.sotaro.io/`[1].

1 Introduction

The rapid increase in the number of scientific publications is observed in various fields (Bornmann and Mutz, 2015), and the field of natural language processing (NLP) is no exception. The main paper repository of NLP, ACL Anthology (Bollmann et al., 2023), has grown its number of stored papers by 70% from 2019 to 2023. Such information overload makes paper discovery for researchers more challenging. Researchers need to spend more time in finding papers relevant to their research interests. To tackle this challenge, the NLP community has developed various methodologies from both a theoretical and an empirical perspective. Automatic research paper summarization aims to produce short texts that encompass the essential information of the paper to allow researchers to grasp quickly overviews (Cachola et al., 2020; Takeshita et al., 2024a). Information extraction methods can provide structure to a collection of papers by extracting keyphrases (Augenstein et al., 2017) or named entities (Jain et al., 2020). From a more practical perspective, various system demonstrations have been developed, putting the research artefacts, e.g., summarization models, together with a user interface (Schopf and Matthes, 2024; Zheng et al., 2024; Lin et al., 2024).

In our previous work, we introduced *GenGO* (Takeshita et al., 2024b)[2], a system where users can retrieve ACL Anthology papers using semantic text encoders enriched with various additional information, such as aspect-based summaries and extracted named entities. While *GenGO* helps researchers quickly discover relevant papers, it has several limitations: (i) lack of query-focused personalized summarization: aspect-based summaries in *GenGO* are generated per paper and do not support user-specific requests such as *Summarize paper X from an efficiency perspective*. (Vig et al., 2022; Su et al., 2021). (ii) no support for multi-document summarization: the system cannot synthesize information across multiple papers, e.g., *Generate an overview of different MT evaluation metrics*. (Fabbri et al., 2019; Cui and Hu, 2021). (iii) no flexible question answering: *GenGO* does not allow users to ask direct questions grounded in the content of papers, such as *Does ROUGE use word overlap?* (Nguyen, 2019).

To tackle these limitations, we present a new system, dubbed *GenGO Ultra*, which uses state-of-the-art large language models (LLMs) to dynamically provide responses to user-provided queries using NLP papers stored in *GenGO*'s database. This solves the three aforementioned limitations of the previous system with one unified user interface. Differently from other running LLM-powered sys-

[1] Demo video: https://youtu.be/6r4CBgHoGLU

[2] `https://gengo.sotaro.io/`

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 242–251
July 27 - August 1, 2025

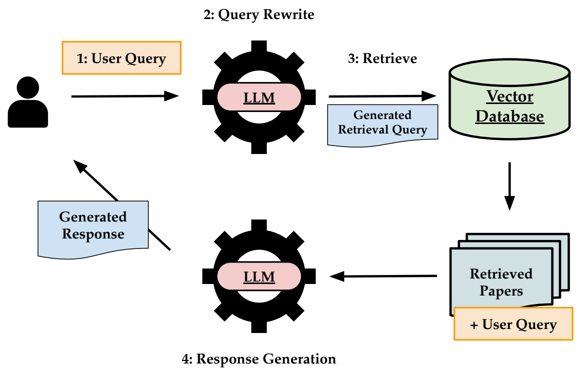

Figure 1: A system overview. Our system first rewrites user-provided query into a retrieval-friendly text which also takes the interaction history into account. This query is then used to retrieve N relevant papers from our vector database (N is set to ten by default but it is adjustable between one and fifteen.) The retrieved papers are fed together with the system prompt and the initial user query to the LLM to produce the response, which is finally presented to the user.

tems, we build our system transparently by using open-weight models and open-sourced system code in which users can examine how the response is generated. Finally, we perform both component-level and end-to-end evaluations to measure system performance with different LLMs.

2 *GenGO* Project

Our present system extends its predecessor system, namely *GenGO* (Takeshita et al., 2024b), a system for NLP researchers to efficiently explore papers published in ACL conferences. It integrates several NLP models to achieve its goal. Each paper is accompanied by three one-sentence summaries which convey the paper's essential information on different aspects (Challenge, Approach, and Outcome) (Takeshita et al., 2024a). We also apply a scientific domain named entity recognizer (Jain et al., 2020) and the field-of-study classifier (Schopf et al., 2023) to attach metadata to papers to enhance search and filtering functionalities. Finally, the system provides a semantic search feature by using a lightweight contrastively trained text encoder.

While these features can improve researchers' paper discovery experience compared to the original paper repository, there are still three major limitations in functionalities that are hindering the system from being more useful. **Dynamic query-focused summarization:** while pre-computed aspect-based summaries can provide a multi-dimensional overview of a paper to enable researchers to quickly understand the essence of the paper, the current system cannot generate a personalized summary for a user-provided query on the fly. **Multi-document summarization:** current system shows summaries for each paper independently, i.e., they cannot provide an overview of a topic in NLP by gathering information from multiple relevant papers. **Flexible QA:** while *GenGO*'s semantic search feature can provide a list of relevant papers given a user query, it cannot directly answer a question using the information from papers.

In the remainder of this paper, we describe how our new system, *GenGO Ultra*, addresses these limitations by integrating LLMs.

3 *GenGO Ultra*

GenGO Ultra is a retrieval augmented generation (RAG) system, i.e., the underlying LLM uses the relevant papers as contexts to generate a response to a user-provided query. By complementing LLMs with retrieval, RAGs can improve LLMs' performance on knowledge-intensive tasks (Lewis et al., 2020) and enable them to incorporate up-to-date information (Ovadia et al., 2024). In our case, it allows us to implemented features that are described in the following section.

3.1 Features

Generation with citations. By prompting the LLM to include references from which the model extracts the information, our system allows users to quickly jump from the generated response to the corresponding paper, enabling researchers to validate the output by reading the source document (Gao et al., 2023; Li et al., 2024).

Collection-specific querying. By default, the system considers the whole collection of papers to respond to the user-provided query, however, it is also possible to query for a specific conference proceeding. This enables users to, for instance, have an overview of a conference they are participating in. To do so, users can first open a conference proceeding in *GenGO*[3] and click the 'Load this conference in *GenGO Ultra*' button.

[3]Example, AACL 2022: `https://gengo.sotaro.io/collections/2022.aacl`

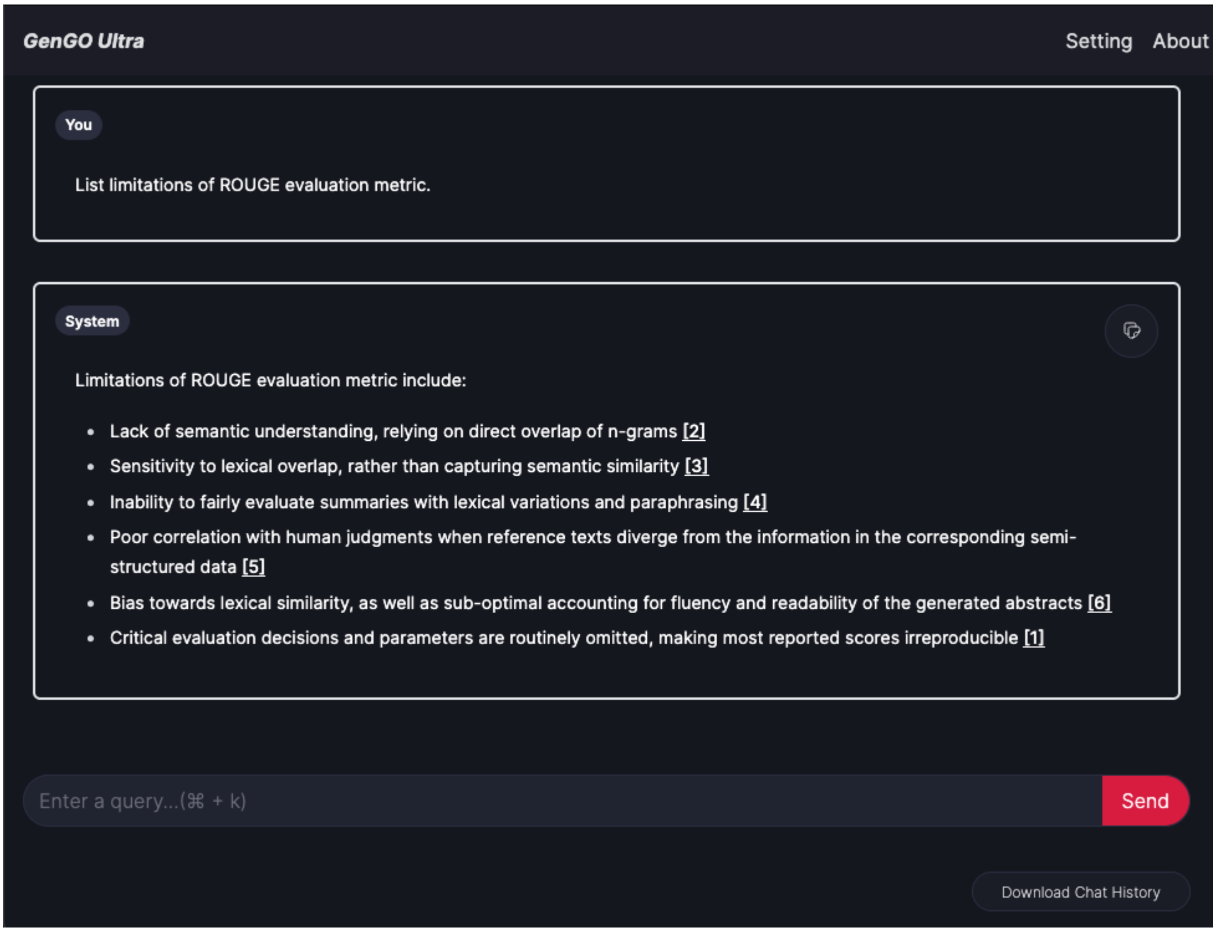

Figure 2: A screenshot of one system-user interaction. *GenGO Ultra* generates a concisely formatted response with references to published papers.

Customizability. Users can choose the underlying LLM from multiple options to enable the qualitative comparison on our system. Currently, users can select from five popular LLMs. We plan to add more models in the future.

Interaction export. Similar existing RAG-based systems hide how the LLMs are provided with different system prompts or the list of contexts fed to the LLM as context, making the response generation process opaque. In our system, users can easily export the entire interaction, including the system prompt as well as the context composed of retrieved papers. This provides transparency to our system and enables users to examine how their queries result in the generated responses.

3.2 System Description

Overview. Fig. 1 shows an overview of our system, composed of two main components in our system, namely an LLM and a vector database.

Query rewriting. Instead of directly using a user query as a search input to retrieve relevant papers, we first re-write it using an LLM similarly as done in Ma et al. (2023). This lets us (i) obtain more search-friendly text, and (ii) take the previous interactions between the system and the user into account. When the user writes a follow-up query regarding the previous interactions like *Tell me more about this from an empirical perspective.*, directly using this as a search query will not return any meaningful results. This re-writing process with the interaction history is required to achieve consistent interaction.

Paper retrieval. Relevant papers are retrieved by computing cosine similarity between paper vectors and search query converted from the user provided query. At the time of writing, we are using a lightweight encoder, snowflake-arctic-embed-s, introduced by Merrick et al. (2024). To store the paper data, we use the same database as the predecessor *GenGO* project in our present system. See more details about the construction of this database

Model	Params (M)	LitSearch		SciDocs		SciFact	
		nDCG@10	MAP@10	nDCG@10	MAP@10	nDCG@10	MAP@10
snowflake-m-v1.5	109	**0.5172**	**0.4764**	**0.2149**	**0.1296**	**0.7472**	0.6689
snowflake-arctic-m	109	0.5124	0.4738	0.2109	0.1269	0.7465	**0.6858**
e5-small-v2	33	0.3781	0.3348	0.1771	0.1031	0.7078	0.6435
bge-small-en-v1.5	33	0.4283	0.3850	0.2164	0.1229	0.7469	0.6681
snowflake-arctic-xs	23	0.4475	0.4110	0.1835	0.1092	0.6769	0.5978
all-MiniLM-L6-v2	23	0.5045	0.3053	0.2309	0.1294	0.6602	0.5959

Table 1: Retrieval performance by six lightweight text encoders on three scientific domain datasets. The performance is measured by nDCG@10 and MAP@10. Higher scores indicate better performance. The best models on each metric and dataset are in **bold**.

		SciTLDR		ACLSum					
				Challenge		Approach		Outcome	
Model	S	R-2	R-K	R-2	R-K	R-2	R-K	R-2	R-K
LL3.3	70	16.2	55.1	9.4	**86.5**	**18.4**	84.8	**13.8**	85.2
LL3.1	8	**16.8**	51.1	7.6	83.4	15.2	85.6	12.1	84.3
Mi 3	24	16.1	50.9	**10.1**	72.7	17.8	83.9	12.5	80.7
Mix	8x22	14.3	57.1	8.1	**86.5**	16.5	**88.3**	12.4	86.7
Mix	8x7	14.2	**58.0**	7.7	82.9	14.7	86.3	12.6	**88.6**

Table 2: Performance of five open-weight LLMs on two summarization datasets. ACLSum is an aspect-based summarization dataset with three aspects. The number of **P**arameters is shown in billions. The **Mix**tral models are based on mixture-of-experts architecture; 8x22 in parameter count means the model has 8 experts with 22 billion parameters each. **LL**, **Mi**, and **Mix** stand for LLaMA, Mistral, and Mixtral, respectively.

in our previous paper (Takeshita et al., 2024b).

Response generation. After the retrieval, we feed the list of relevant papers to an LLM together with the original user query and our system prompt. Our current system prompt covers the following instructions in its essence: the final response must (i) be concise and accurate, (ii) cite the relevant papers from the context, (iii) be contained within 150 words, (iv) use the markdown syntax, (v) not contain URLs or links. See Table 7 for our full system prompt. While users can select from multiple LLMs, by default, our system uses LLaMA 3.3 with 70B parameters from Meta[4]. Our LLMs are hosted using Together AI[5].

4 Evaluation

In this section, we evaluate six text encoders on three paper retrieval datasets (§4.1), and five LLMs on paper summarization and instruction-following tasks (§4.1), and their combinations on end-to-end response generation task (§4.2).

4.1 Component-level evaluation

Retrieval. We evaluate the retrieval performance of six text encoders (Merrick et al., 2024; Wang et al., 2022; Xiao et al., 2023)[6], on three scientific domain datasets (Cohan et al., 2020; Wadden et al., 2020; Ajith et al., 2024). All models are small compared to the current state-of-the-art text encoders such as E5-Mistral by Wang et al. (2024). This is because we encode the query text on the user's device (e.g., laptop or smartphone), where computational resources are limited. We take this on-device encoding approach to reduce our cost to run the system (i.e., we do not need to send the query text to hosted APIs that require fees). More specifically, two models have 109 million parameters, and the other four have fewer than 33 million parameters. The results are shown in Table 1. Between the two larger models, snowflake-m-v1.5 outperforms the other model in almost all cases, and we observe a large performance gap between the larger models and the smaller models. As it is still possible to run 109M parameter models on mobile devices, we currently opt for the snowflake-m-v1.5.

Summarization. While our system mainly aims to provide multi-document summarization functionality, due to the lack of high-quality multi-document summarization datasets in the scientific domain, as a proxy assessment, we evaluate a set of five open-weight LLMs on two single-document summarization datasets, namely SciTLDR (Cachola et al., 2020) and ACLSum (Takeshita et al., 2024a). The former contains pairs of paper abstracts from machine learning conferences and one-sentence summaries written by paper authors. The

[4]https://github.com/meta-llama/llama-models/blob/main/models/llama3_3/MODEL_CARD.md

[5]https://www.together.ai/

[6]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Model	Params (B)	T1	T2	T3	T4	T5	T5'	T6	T6'	T7	T8	T8'	Avg
LL3.3	70	48.0	**21.9**	62.3	**33.5**	**83.1**	**69.5**	51.0	**52.9**	**28.0**	**47.5**	35.4	**48.5**
LL3.1	8	46.4	13.0	42.2	21.1	69.2	54.1	53.0	46.2	5.3	43.0	**41.2**	39.5
Mi 3	24	**52.3**	16.8	**63.6**	26.9	79.1	59.7	**58.0**	49.9	0.9	41.5	30.9	43.6
Mix	8x7	46.3	13.7	43.9	18.1	71.7	54.4	52.0	45.6	18.2	38.1	26.1	38.9

Table 3: Performance of instruction-following ability evaluated on SciRIFF benchmark. The complete names of tasks and the corresponding papers are listed in Table 6 in the Appendix. Differently from our other evaluations of LLMs, Mixtral 8x22B is omimited due to its large memory consumption and the long context of tasks in the benchmark.

Model	Params (B)	Coh	Con	Flu	Rel
LL3.3	70	**3.54**	**3.40**	2.56	**3.08**
LL3.1	8	3.52	2.36	**2.83**	2.77
Mi 3	24	2.96	2.48	2.33	2.50
Mix	8x22	1.20	2.48	2.74	2.74
Mix	8x7	1.11	1.20	2.43	1.23

Table 4: Results of end-to-end evaluation. We use the quantized Qwen2.5-32B-Instruct as the evaluator, and the evaluation prompt is based on Liu et al. (2023).

latter is an aspect-based summarization dataset where each data point is composed of the paper content and three sentences summarizing the corresponding paper from different perspectives (Challenge, Approach, and Conclusion). We use two evaluation metrics, namely ROUGE-2 (Lin, 2004) and its keyword-oriented extension, ROUGE-K (Takeshita et al., 2024c). We list the evaluated LLMs in Table 5 in the Appendix. The results of our summarization evaluation are shown in Table 2. While LLaMA 3.3 marks the highest number of best scores among the five models, the results are mixed, and it is hard to determine the best-performing model in this experiment. However, interestingly, models from the LLaMA family outperform the Mistral family on all the datasets when measured by ROUGE-2, and the result is the opposite on ROUGE-K, i.e., Mistral models are better at including more keywords than LLaMA counterparts.

LLM Instruction-following General-purpose LLMs often lack domain-specific scientific knowledge and may not be well-suited for scientific tasks (Li et al., 2025). To identify models capable of handling instruction-following tasks relevant to researchers, we perform evaluation using the SciRIFF benchmark (Wadden et al., 2024). SciRIFF is a collection of diverse tasks spanning multiple scientific domains, with human-annotated inputs and outputs. Successfully completing these tasks requires models to reason over long input contexts, making this benchmark suitable for our interests. We select 4,622 samples covering 8 tasks that require structured output in JSON format. In preliminary experiments, we observed that many incorrect predictions resulted from parsing errors caused by free-form output. By enforcing a specified format through constrained decoding, we significantly reduced the number of invalid JSON outputs. To achieve such constrained generation, we make use of outlines introduced by Willard and Louf (2023). This adjustment allows for a more accurate assessment of a model's ability to follow instructions. LLaMA 3.3 achieves the highest average performance across all tasks. This result encourages us to set it as the default LLM in our system.

4.2 End-to-end evaluation

While our previous experiments evaluate LLMs and encoders individually, in this section, we aim to evaluate our RAG system as a whole with different LLMs. To this end, we employ LLM-as-a-judge as our evaluation strategy, where the output from the system is evaluated automatically by an LLM (Liu et al., 2023). Although there are works which report biases in this evaluation schema (Raina et al., 2024; Chen et al., 2024), we opt for this evaluation framework due to the lack of suitable existing datasets and the financial resources to perform more robust evaluation, such as manual evaluation (Chen et al., 2024; Chiang and Lee, 2023). To reduce one of the issues currently known for this evaluation strategy, namely self-preference bias (Liu et al., 2024), our evaluator LLM (Team, 2024) is not one of our considered models. For the prompting strategy, we take the prompt used by Liu et al. (2023), which instructs an evaluator LLM to assess model outputs on four aspects, namely coherence, consistency, fluency, and relevance. We constructed a dataset composed of 25 questions and responses generated by the targeted models for this evaluation. Table 4 shows the results. Contrary to the summarization evaluation in §4.1, we observe a clear dominance of LLaMA 3.3, the model with the largest active parameter size. Given this and the results from the previous instruction-following experiments in §4.1, we set LLaMA 3.3 as the default LLM in *GenGO-Ultra*, however, users can change to the other four models in the setting page.

5 Limitations

While we believe *GenGO Ultra* can assist NLP researchers to efficiently explore published papers, there are some limitations. (i) limited instruction-following ability: we observe that the system sometimes does not fully capture the intent of the instruction, which is also observed with more powerful proprietary models (Wadden et al., 2024). (ii) hallucination: in some cases, even when the context papers do not provide relevant information to the user query, LLMs still generate an answer with claims that are not present in cited papers. (iii) retrieval performance: the current system does not implement the most powerful text encoders (Wang et al., 2024) and iterative retrieval strategies (Shao et al., 2023) due to the limited computation resources. (iv) limited LLM availability: due to the limited budget, we set a monthly upper limit on LLM usage, after which our system shuts down until the beginning of the following month.

While the last two points are inevitable due to our limited resources, we plan to improve our system in the first two points by incorporating advanced LLM prompting strategies. Kirstein et al. (2025) propose a multi-LLM framework where two LLMs assess and provide feedback so that the response-generating LLM can iteratively improve its output quality. To combat the hallucination problem, Dhuliawala et al. (2024) introduce a multi-step prompting pipeline composed of drafting and verifying steps. The authors show that this approach helps to reduce hallucination by LLMs on various tasks, including longform text generation.

6 Conclusion

In this paper, we described *GenGO Ultra*, a RAG system which enables NLP researchers to have flexible interactions with publications to foster an efficient literature search. It effectively connects LLMs to our publication vector database as a source of NLP knowledge to enhance the LLM's ability to achieve flexible interactions. We also performed a series of model evaluations on different granularities and tasks to determine the most suitable sets of NLP models for our system.

Acknowledgements

We acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG. We thank our colleagues Daniel Ruffinelli and Tommaso Green for their comments on a draft of this paper.

References

Anirudh Ajith, Mengzhou Xia, Alexis Chevalier, Tanya Goyal, Danqi Chen, and Tianyu Gao. 2024. LitSearch: A retrieval benchmark for scientific literature search. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15068–15083, Miami, Florida, USA. Association for Computational Linguistics.

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555, Vancouver, Canada. Association for Computational Linguistics.

Marcel Bollmann, Nathan Schneider, Arne Köhn, and Matt Post. 2023. Two decades of the ACL Anthology: Development, impact, and open challenges. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 83–94, Singapore, Singapore. Empirical Methods in Natural Language Processing.

Lutz Bornmann and Rüdiger Mutz. 2015. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *J. Assoc. Inf. Sci. Technol.*, 66(11):2215–2222. Publisher: Wiley.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel Weld. 2020. TLDR: Extreme summarization of scientific documents. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777, Online. Association for Computational Linguistics.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA. Association for Computational Linguistics.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Peng Cui and Le Hu. 2021. Topic-guided abstractive multi-document summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1463–1472, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jay DeYoung, Eric Lehman, Benjamin Nye, Iain Marshall, and Byron C. Wallace. 2020. Evidence inference 2.0: More data, better models. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 123–132, Online. Association for Computational Linguistics.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

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*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

Frederic Thomas Kirstein, Terry Lima Ruas, and Bela Gipp. 2025. What‘s wrong? refining meeting summaries with LLM feedback. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2100–2120, Abu Dhabi, UAE. Association for Computational Linguistics.

Anne Lauscher, Brandon Ko, Bailey Kuehl, Sophie Johnson, Arman Cohan, David Jurgens, and Kyle Lo. 2022. MultiCite: Modeling realistic citations requires moving beyond the single-sentence single-label setting. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1875–1889, Seattle, United States. Association for Computational Linguistics.

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

Sihang Li, Jin Huang, Jiaxi Zhuang, Yaorui Shi, Xiaochen Cai, Mingjun Xu, Xiang Wang, Linfeng Zhang, Guolin Ke, and Hengxing Cai. 2025. SciLitLLM: How to adapt LLMs for scientific literature understanding. In *The Thirteenth International Conference on Learning Representations*.

Weitao Li, Junkai Li, Weizhi Ma, and Yang Liu. 2024. Citation-enhanced generation for LLM-based chatbots. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1451–1466, Bangkok, Thailand. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Guanyu Lin, Tao Feng, Pengrui Han, Ge Liu, and Jiaxuan You. 2024. Arxiv copilot: A self-evolving and efficient LLM system for personalized academic assistance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 122–130, Miami, Florida, USA. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Yiqi Liu, Nafise Moosavi, and Chenghua Lin. 2024. LLMs as narcissistic evaluators: When ego inflates evaluation scores. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12688–12701, Bangkok, Thailand. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language

Processing, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315, Singapore. Association for Computational Linguistics.

Luke Merrick, Danmei Xu, Gaurav Nuti, and Daniel Campos. 2024. Arctic-embed: Scalable, efficient, and accurate text embedding models. *arXiv preprint arXiv:2405.05374*.

Vincent Nguyen. 2019. Question answering in the biomedical domain. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 54–63, Florence, Italy. Association for Computational Linguistics.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2024. Fine-tuning or retrieval? comparing knowledge injection in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 237–250, Miami, Florida, USA. Association for Computational Linguistics.

Vyas Raina, Adian Liusie, and Mark Gales. 2024. Is LLM-as-a-judge robust? investigating universal adversarial attacks on zero-shot LLM assessment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7499–7517, Miami, Florida, USA. Association for Computational Linguistics.

Arkadiy Saakyan, Tuhin Chakrabarty, and Smaranda Muresan. 2021. COVID-fact: Fact extraction and verification of real-world claims on COVID-19 pandemic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2116–2129, Online. Association for Computational Linguistics.

Mourad Sarrouti, Asma Ben Abacha, Yassine Mrabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tim Schopf, Karim Arabi, and Florian Matthes. 2023. Exploring the landscape of natural language processing research. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1034–1045, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Tim Schopf and Florian Matthes. 2024. NLP-KG: A system for exploratory search of scientific literature in natural language processing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 127–135, Bangkok, Thailand. Association for Computational Linguistics.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274, Singapore. Association for Computational Linguistics.

Dan Su, Tiezheng Yu, and Pascale Fung. 2021. Improve query focused abstractive summarization by incorporating answer relevance. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3124–3131, Online. Association for Computational Linguistics.

Sotaro Takeshita, Tommaso Green, Ines Reinig, Kai Eckert, and Simone Ponzetto. 2024a. ACLSum: A new dataset for aspect-based summarization of scientific publications. 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 6660–6675, Mexico City, Mexico. Association for Computational Linguistics.

Sotaro Takeshita, Simone Ponzetto, and Kai Eckert. 2024b. GenGO: ACL paper explorer with semantic features. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 117–126, Bangkok, Thailand. Association for Computational Linguistics.

Sotaro Takeshita, Simone Ponzetto, and Kai Eckert. 2024c. ROUGE-k: Do your summaries have keywords? In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 69–79, Mexico City, Mexico. Association for Computational Linguistics.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16:1–28.

Jesse Vig, Alexander Fabbri, Wojciech Kryscinski, Chien-Sheng Wu, and Wenhao Liu. 2022. Exploring neural models for query-focused summarization. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1455–1468, Seattle, United States. Association for Computational Linguistics.

Vijay Viswanathan, Luyu Gao, Tongshuang Wu, Pengfei Liu, and Graham Neubig. 2023. DataFinder: Scientific dataset recommendation from natural language descriptions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10288–10303, Toronto, Canada. Association for Computational Linguistics.

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.

David Wadden, Kejian Shi, Jacob Morrison, Aakanksha Naik, Shruti Singh, Nitzan Barzilay, Kyle Lo, Tom Hope, Luca Soldaini, Zejiang Shen, Doug Downey, Hannaneh Hajishirzi, and Arman Cohan. 2024. SciRIFF: A resource to enhance language model instruction-following over scientific literature. In *Neurips 2024 Workshop Foundation Models for Science: Progress, Opportunities, and Challenges*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand. Association for Computational Linguistics.

Brandon T Willard and Rémi Louf. 2023. Efficient guided generation for llms. *arXiv preprint arXiv:2307.09702*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding.

Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin, Binjie Wang, Yun Luo, Renjie Pan, Yang Xu, Qingkai Min, Zizhao Zhang, Yiwen Wang, Wenjie Li, and Pengfei Liu. 2024. OpenResearcher: Unleashing AI for accelerated scientific research. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 209–218, Miami, Florida, USA. Association for Computational Linguistics.

A Appendix

Name	Licence	URL
meta-llama/Llama-3.3-70B-Instruct	Llama 3.3 Community License	https://huggingface.co/meta-llama/Llama-3...
meta-llama/Llama-3.1-8B-Instruct	Llama 3.1 Community License	https://huggingface.co/meta-llama/Llama-3....
mistralai/Mistral-Small-24B-Instruct-2501	Apache license 2.0	https://huggingface.co/mistralai/Mistral-Small...
mistralai/Mixtral-8x22B-Instruct-v0.1	Apache license 2.0	https://huggingface.co/mistralai/Mixtral-8x22B...
mistralai/Mixtral-8x7B-Instruct-v0.1	Apache license 2.0	https://huggingface.co/mistralai/Mixtral-8x7B...

Table 5: List of LLMs from our experiments with their licenses and URLs.

Task ID	Task Name	Evaluation Metric	Publication
T1	BioASQ	exact F1	Tsatsaronis et al. (2015)
T2	Evidence Inference	string overlap approximate F1	DeYoung et al. (2020)
T3	MultiCite	exact F1	Lauscher et al. (2022)
T4	SciERC (NER)	exact F1	Luan et al. (2018)
T5	SciFact entailment	evidence token F1	Wadden et al. (2020)
T5'	SciFact entailment	label F1	Wadden et al. (2020)
T6	CovidFact entailment	evidence token F1	Saakyan et al. (2021)
T6'	CovidFact entailment	label F1	Saakyan et al. (2021)
T7	DataFinder	exact F1	Viswanathan et al. (2023)
T8	HealthVer	evidence token F1	Sarrouti et al. (2021)
T8'	HealthVer	label F1	Sarrouti et al. (2021)

Table 6: List of datasets used in our instruction-following evaluation.

You are a helpful search assistant.
Your task is to deliver a concise and accurate response to a user's query, drawing from the given research papers.
Your answer must be precise, of high-quality, and written by an expert using an unbiased and journalistic tone.
It is EXTREMELY IMPORTANT to directly answer the query. NEVER say 'based on the search results' or start your answer with a heading or title.
Get straight to the point.
Your answer MUST be less than 150 words.

You MUST cite the relevant papers that answer the query.
Use PUIDs to cite the relevant papers AT THE END of a sentence.
Do not mention any irrelevant papers.
You MUST ADHERE to the following instructions for citing papers:
to cite a paper, enclose relevant paper's PUIDs at the end of the output sentence, like '(PUID:1)(PUID:3)'
NO SPACE between the last word and the citation, and ALWAYS use brackets. Only use this format with PUIDs to cite search results.
DO NOT write a References section.
Ignore the papers that are not relevant to the query.
You MUST ADHERE to the following formatting instructions:
Use headings level 2 and 3 to separate sections of your response, like '## Header', but NEVER start an answer with a heading or title of any kind (i.e. Never start with #).
Use single new lines for lists and double new lines for paragraphs.
NEVER write URLs or links.

Research papers:
<Relevant Papers>

Query: <User-provided Query>

Use markdown list to structure the output.
Make sure to cite relevant papers using PUIDs, like '(PUID:1)(PUID:3)'.
Do not include reference section at the end.

Table 7: System prompt used to generate the response the user query using retrieved papers.