# NLP-KG: A System for Exploratory Search of Scientific Literature in Natural Language Processing

**Tim Schopf** and **Florian Matthes**
Technical University of Munich, Department of Computer Science, Germany
{tim.schopf,matthes}@tum.de

## Abstract

Scientific literature searches are often exploratory, whereby users are not yet familiar with a particular field or concept but are interested in learning more about it. However, existing systems for scientific literature search are typically tailored to keyword-based lookup searches, limiting the possibilities for exploration. We propose NLP-KG, a feature-rich system designed to support the exploration of research literature in unfamiliar natural language processing (NLP) fields. In addition to a semantic search, NLP-KG allows users to easily find survey papers that provide a quick introduction to a field of interest. Further, a Fields of Study hierarchy graph enables users to familiarize themselves with a field and its related areas. Finally, a chat interface allows users to ask questions about unfamiliar concepts or specific articles in NLP and obtain answers grounded in knowledge retrieved from scientific publications. Our system provides users with comprehensive exploration possibilities, supporting them in investigating the relationships between different fields, understanding unfamiliar concepts in NLP, and finding relevant research literature. Demo, video, and code are available at: https://github.com/NLP-Knowledge-Graph/NLP-KG-WebApp.

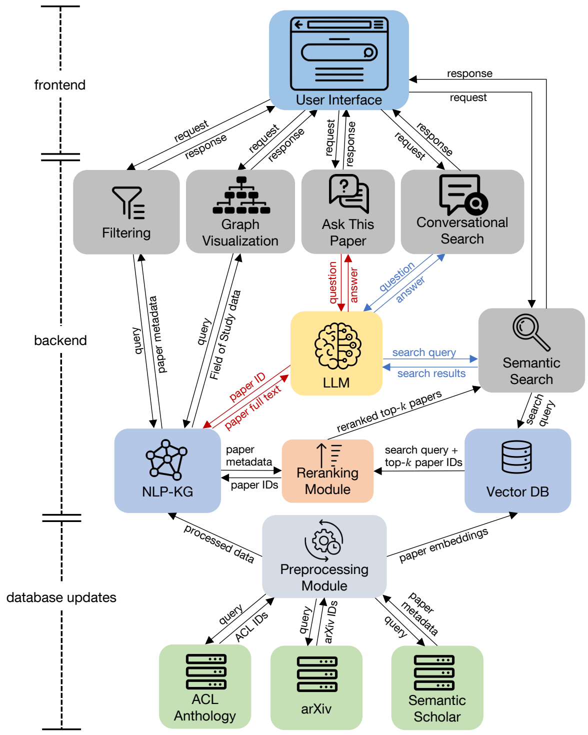


Figure 1: The architecture of our system. The direction of an arrow represents the direction of data flow. The red arrows show how the autoregressive Large Language Model (LLM) routes the data for the *Ask This Paper* feature, while the blue arrows show how the LLM routes the data for the *Conversational Search* feature. The preprocessing module regularly fetches new publications and processes them to update the knowledge graph and the vector database.

## 1 Introduction

The body of natural language processing (NLP) literature has experienced remarkable growth in recent years, with articles on various topics and applications being published in an increasing number of journals and conferences (Schopf et al., 2023). To browse and search the increasing amount of NLP-related literature, researchers may use systems such as Google Scholar[1] or Semantic Scholar (Kinney et al., 2023). Both systems cover a wide variety of academic disciplines. Although this has advantages, the lack of focus on NLP literature also has disadvantages, e.g., the potential to retrieve lots of search results containing many irrelevant papers (Mohammad, 2020). For example, when interested in NLP literature on *emotion* or *privacy*, searching for it on Google Scholar is less efficient than searching for it on a platform dedicated to NLP literature. Further, scholarly literature searches are often exploratory, whereby users are not yet familiar with a particular field or concept and are interested in learning more about it (Soufan et al., 2022). How-

[1] https://scholar.google.com



*Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 127–135
August 11-16, 2024 ©2024 Association for Computational Linguistics

ever, commonly used search systems are usually optimized for targeted lookup searches, limiting search and exploration to keyword-based searches and citation-based exploration.

In this paper, we present a system to support the exploration of NLP research literature from unfamiliar fields using a knowledge graph (KG) and state-of-the-art retrieval approaches. Our main contributions comprise the following features:

- **Graph visualization** of hierarchically structured Fields of Study (FoS) in NLP. FoS are academic disciplines and concepts, commonly comprised of (but not limited to) tasks or methods (Shen et al., 2018). The graph visualization offers researchers new to a field a starting point for their exploration and supports them to familiarize themselves with a field and its related areas.

- **Semantic search** provides a familiar interface to enable keyword-based searches for publications, authors, venues, and FoS in NLP.

- **Conversational search** responds to NLP-related user questions in natural language and grounds the answers in knowledge from academic publications using a Retrieval Augmented Generation (RAG) pipeline. This feature allows users to ask questions about unfamiliar concepts and fields in NLP and provides explanations as well as reference literature for further exploration.

- **Ask this paper** uses an autoregressive Large Language Model (LLM) to answer in-depth user questions about specific publications based on their full texts. This can support users to understand papers from unfamiliar fields.

- **Advanced filters** can filter the search results for specific FoS, venues, dates, citation counts, or survey papers. Especially filtering by survey papers can support users to quickly get an introduction to their field of interest.

Our system is not intended to replace commonly used search engines but to serve as a supplementary tool for dedicated exploratory search of NLP research literature.

## 2 Related Work

Google Scholar, Semantic Scholar (Kinney et al., 2023), ArnetMiner (Tang et al., 2008), Microsoft Academic Graph (MAG) (Sinha et al., 2015; Wang et al., 2020), OpenAlex (Priem et al., 2022), and Open Research Knowledge Graph (ORKG) (Jaradeh et al., 2019; Auer et al., 2020) are all systems for search and discovery of academic literature covering a wide range of scholarly domains.

Weitz and Schäfer (2012) focus on citation analyses of NLP-related literature. CL Scholar (Singh et al., 2018) is a system that can answer binary, statistical, and list-based queries about computational linguistics publications. Additionally, NLP Scholar (Mohammad, 2020) provides interactive visualizations of venues, authors, n-grams, and keywords extracted from NLP-related publications, while the NLP Explorer (Parmar et al., 2020) provides FoS tags and temporal statistics to search and explore the field of NLP.

## 3 NLP-KG

A well-organized hierarchical structure of FoS and an accurate mapping between these FoS and scholarly publications can enable a streamlined and satisfactory exploration experience (Shen et al., 2018). Further, semantic relations between scholarly entities can be easily modeled in a graph representation. Therefore, we construct the Natural Language Processing Knowledge Graph (NLP-KG) as the core of our system that links FoS, publications, authors, and venues via semantic relations. In addition, we integrate a LLM in our retrieval pipeline that can enhance the exploration experience by providing accurate responses to user queries (Zhu et al., 2024). Figure 1 illustrates how the knowledge graph and the LLM are integrated into our system.

### 3.1 Fields of Study Hierarchy Construction

During exploration, users typically navigate from more well-known general concepts to less well-known and more specific concepts. Therefore, we use a semi-automated approach to construct a high-quality, hierarchical, acyclic graph of FoS in NLP. As a starting point, we use a readily available high-level taxonomy of concepts in NLP (Schopf et al., 2023). At the top level, this NLP taxonomy includes 12 different concepts covering the wide range of NLP, and consequently, additional concepts can be considered as hyponyms thereof. In total, this NLP taxonomy already includes 82 different FoS, to which we subsequently add further FoS as hyponyms and co-hyponyms.

**Automated Knowledge Extraction** For automated extraction of FoS and hierarchical relations, we use a corpus of titles and abstracts of research publications from the ACL Anthology[2] and the cs.CL category of arXiv[3]. After removing duplicates, the corpus includes a total of 116,053 documents. For entity and relation extraction, we fine-tune Packed Levitated Marker (PL-Marker) models (Ye et al., 2022) on a slightly adapted SciERC dataset (Luan et al., 2018). Since we do not distinguish between different entity types in our FoS hierarchy graph, we process the SciERC dataset to unify all entity types and transform the original named entity recognition task into a more simple entity extraction task. Additionally, we only use the *Hyponym-of* relationship to extract hierarchical relations. Finally, we experiment with BERT (Devlin et al., 2019), SciBERT (Beltagy et al., 2019), SPECTER2 (Singh et al., 2023), and SciNCL (Ostendorff et al., 2022) as base models.

| Task → | Entity Extraction | | | Relation Extraction | | |
|---|---|---|---|---|---|---|
| **Model ↓** | **P** | **R** | **$F_1$** | **P** | **R** | **$F_1$** |
| BERT | 68.87 | 66.63 | 67.73 | 70.01 | 68.28 | 69.13 |
| SciBERT | 69.91 | **67.09** | **68.47** | 71.23 | **69.63** | **70.42** |
| SPECTER2 | **69.99** | 66.52 | 68.21 | 69.66 | 68.95 | 69.30 |
| SciNCL | 69.59 | 65.39 | 67.42 | **71.24** | 68.28 | 69.73 |

Table 1: Evaluation results for PL-Marker fine-tuning on the processed SciERC test set using different base models. We report micro (P)recision, (R)ecall, and $F_1$ scores.

The evaluation results for PL-Marker fine-tuning are shown in Table 1. Based on these results, we select the SciBERT-based PL-Marker models to extract entities and relations from our corpus of NLP-related research articles, resulting in large sets of entities and relations. To resolve duplicate entities, we use a rule-based approach that recognizes synonyms by unifying special characters and extracting abbreviations of terms that appear in parentheses immediately following an entity. In order to limit the set of eligible entities and relationships to high-quality ones, we select only those that are extracted more frequently than the thresholds of $t_{entities} = 100$ and $t_{relations} = 3$.

**Manual Correction & Construction** The extracted entities and relationships are passed to domain experts for validation and correction. In this case, the authors of the present work act as domain experts. If the domain experts consider a candidate triplet valid, it is manually inserted into the FoS hierarchy graph at the correct position. Otherwise, the candidate triplet is corrected, if possible, and only then inserted. Some candidate triples cannot be corrected since they involve out-of-domain terms, e.g., from the legal or medical field, and are, therefore, intentionally disregarded. Finally, we use GPT-4 (OpenAI, 2023) to generate short textual descriptions for each FoS. Table 2 shows an overview of the resulting FoS hierarchy graph.

| # Fields of Study | # Relations | Max Depth |
|---|---|---|
| 421 | 530 | 7 Levels |

Table 2: Overview of the resulting FoS hierarchy graph.

### 3.2 Fields of Study Classification

To automatically assign research publications to the corresponding FoS in the hierarchy graph, we use a two-step classification approach. In the first step, we use the fine-tuned classification model of Schopf et al. (2023). It achieves an $F_1$ score of 93.21, using the 82 high-level FoS of the NLP taxonomy as classes, which we use as the starting point for our hierarchy graph.

In the second step, we use the remaining FoS of our hierarchy graph as classes. Since we do not have sufficient annotated data to train a well-performing classifier, we use a rule-based approach. Thereby, publications are assigned to FoS depending on whether the stemmed FoS names or their stemmed synonyms are contained in the stemmed publication titles.

### 3.3 Survey Paper Classification

To enable filtering by survey papers, we train a binary classifier that can automatically classify research publications into surveys and non-surveys. To this end, we construct a new dataset of survey and non-survey publications in NLP. We obtain a list of candidate survey publications from keyword-based searches in the ACL Anthology and the arXiv cs.CL category using search terms such as *"survey"*, *"a review"*, or *"landscape"*. We then manually annotate the candidate publications as positives if we consider them to be surveys based on their titles and abstracts. For negative sampling, we use the corpus of NLP-related publications described in §3.1, excluding the previously identified positive examples. From this corpus, we randomly sample 15 times the number of positives as negatives to

[2]https://aclanthology.org
[3]https://arxiv.org

account for the inherent under-representation of surveys in conferences and journals. This annotation process results in a dataset of 787 survey and 11,805 non-survey publications in NLP.

Using this survey dataset, we fine-tune and evaluate BERT, SciBERT, SPECTER2, and SciNCL models for binary classification. We create three different stratified 80/20 train/test splits and train all models for two epochs. Following the evaluation results in Table 3, we select the SciNCL-based model as our final classifier.

| Model ↓ | Precision | Recall | $F_1$ | Accuracy |
|---|---|---|---|---|
| BERT | **84.35**±**3.45** | 77.49±5.92 | 80.60±2.07 | 97.68±0.15 |
| SciBERT | 83.32±2.21 | 82.38±1.84 | 82.82±0.81 | 97.87±0.12 |
| SPECTER2 | 82.13±4.58 | 85.77±5.34 | 83.72±0.38 | 97.92±0.08 |
| SciNCL | 82.38±4.01 | **86.53**±**1.74** | **84.35**±**1.67** | **98.04**±**0.22** |

Table 3: Evaluation results for survey paper classification as means and standard deviations on three runs over different random train/test splits. Since the distribution of classes is very unbalanced, we report micro scores.

### 3.4 Additional Metadata

To construct the NLP-KG, we additionally use metadata obtained from the Semantic Scholar API. This includes short one-sentence summaries of publications (TLDRs), SPECTER2 embeddings of publications, author information, as well as citations and references. Further, we use PaperMage (Lo et al., 2023) to obtain the full texts of open-access publications.

### 3.5 Semantic Search

For semantic search, we use a hybrid approach that combines sparse and dense text representations to find the top-$k$ most relevant publications for a query. To this end, the results of BM25 (Robertson and Walker, 1994) and SPECTER2 embedding-based retrieval are merged using Reciprocal Rank Fusion (RRF) (Cormack et al., 2009). To give more weight to the embedding-based approach, we set the $\alpha$ parameter determining the weight between sparse and dense retrieval to 0.8. In addition, we use the S2Ranker (Feldman, 2020) to rerank the top $k = 2000$ retrieved publications using additional metadata from the NLP-KG, such as the number of citations and the publication date.

### 3.6 Conversational Search

To answer NLP-related user questions and recommend relevant literature, we use the LLM in a RAG pipeline. Upon receiving a new user query, the LLM generates search terms using both the query and a one-shot example. These terms are then used for retrieving relevant publications via the semantic search module. Subsequently, the full texts of the top five search results are fed back to the LLM, which generates a response grounded in the retrieved literature. To make the generated answer verifiable for users and denote the knowledge sources, the LLM also generates inline citations. For follow-up queries, the LLM autonomously determines whether to respond using already retrieved publications or to initiate a new search. To reduce the hardware requirements of our server, we use the GPT-4 API for the conversational search and the *Ask This Paper* feature.

### 3.7 Ask This Paper

In addition to the conversational search, the LLM integration enables user inquiries on specific publications via a popup window on each publication page. Users can either pose their own questions or choose from three predefined ones. Using the full text of the publication, the LLM generates verifiable answers supplemented by supporting statements, including section and page references from the publication text. Subsequently, the LLM generates three unique follow-up questions based on the conversation history.

## 4 Demonstration

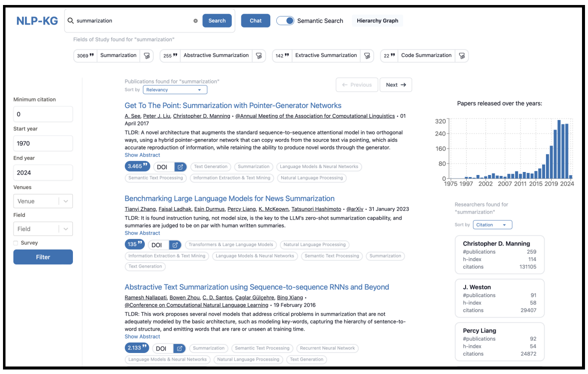


Figure 2: Screenshot showing the semantic search and filtering features.

Our web application is built with Next.js[4] and uses Python[5] for the semantic search and preprocessing modules. The NLP-KG is stored in Neo4j[6] and the embeddings are stored in Weaviate[7]. Our

[4] https://nextjs.org
[5] https://www.python.org
[6] https://neo4j.com
[7] https://weaviate.io


Figure 3: Screenshot of the FoS view and the hierarchy graph visualization.

databases encompass publications from the entire ACL Anthology and the arXiv cs.CL category, enriched with metadata from Semantic Scholar. As illustrated in Figure 1, the preprocessing module regularly fetches new publications, classifies them, and updates our databases.

Figure 2 shows the semantic search interface, allowing users to search for publications, authors, venues, and FoS using keywords via the top search bar. The central area shows retrieved publications, while relevant authors are listed on the right-hand side. Additionally, the top right corner showcases the annual publication count among the search results. On the left-hand side, users can access various filtering options, including the ability to filter by survey publications. Further, a list of FoS related to the search results is displayed at the top of the page, enabling users to navigate to dedicated FoS pages.

Figure 3 shows the FoS page, featuring a brief description of the respective FoS at the top, along with statistics on the annual publication count. The top right corner showcases a relevant section of the FoS hierarchy, enabling exploration of related

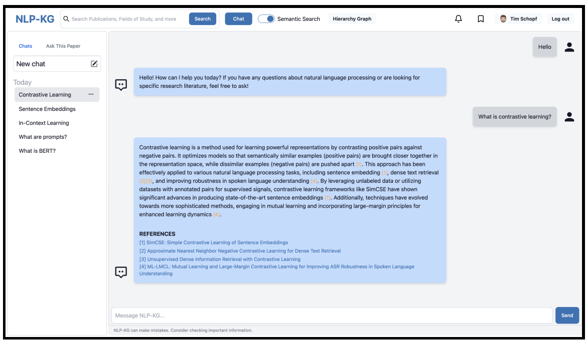


Figure 4: Screenshot of the conversational search feature.

fields. At the bottom of the page, users can explore and filter relevant authors and articles published on this topic.

Figure 4 shows the conversational search feature. Users can pose NLP-related questions to the LLM, which generates responses utilizing knowledge obtained from retrieved publications, accompanied by reference information. To enhance usability, the web application provides clickable links to referenced papers. Additionally, users can conveniently access their conversation history on the left-hand side.

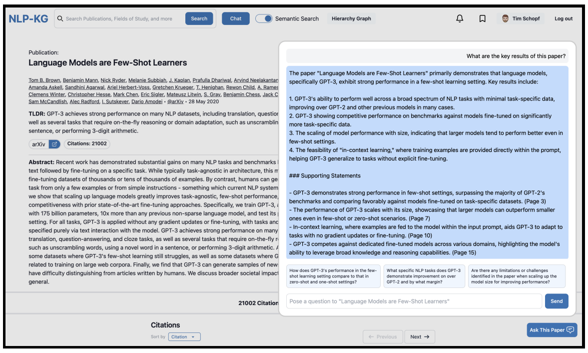


Figure 5: Screenshot of the publication view and the *Ask This Paper* feature.

Figure 5 shows the *Ask This Paper* feature, enabling users to inquire about a specific publication. Accessible via a popup window at each publication page, users can choose from predefined questions or ask custom questions using the input field at the bottom of the chat window.

## 5 Evaluation

### 5.1 Fields of Study Hierarchy Graph

To evaluate the correctness of the FoS hierarchy graph, we conduct a user study involving ten NLP researchers at the PhD level. Participants list five NLP concepts related to their expertise while we ensure their presence in our graph. Subsequently, participants are presented with a visual representation of the constructed graph, initially showing only the first level of FoS in the hierarchy. This requires participants to expand the view by clicking to show the related FoS. Participants are then tasked with locating their provided FoS in the fewest steps possible, with each click or view extension counting as one step. Since the participants selected the FoS for the search themselves, we ensure their familiarity with the target field and related fields. We observe and count every step of the participants throughout their search process. Upon locating their FoS, participants evaluate the correctness of the relations utilized during their navigation and determine potential missing relations. Based on this assessment, we compute Precision, Recall, and $F_1$ scores, as shown in Table 4, to evaluate the correctness of the traversed relations.

Furthermore, we use Mean Absolute Percentage Error (MAPE) to measure the percentage of errors or extra steps that participants make as they navigate the graph to reach their target FoS. We adopt the MAPE metric as follows:

$$\text{MAPE} = \frac{1}{n} \sum \left| \frac{\text{Total \#Steps - Ideal \#Steps}}{\text{Ideal \#Steps}} \right|, \quad (1)$$

where $n = 50$ denotes the number of FoS searches over all participants. In this context, a lower score means that, on average, users were able to find their target FoS with fewer extra steps. For example, a score of zero would mean that each user was able to find their target FoS with the optimal number of steps. Table 4 shows the evaluation results that demonstrate the high quality of the FoS hierarchy graph.

| Precision | Recall | $F_1$ | MAPE |
|---|---|---|---|
| 99.95 | 99.65 | 99.80 | 0.478 |

Table 4: Results for evaluating the correctness of relations in the FoS hierarchy graph.

### 5.2 RAG Performance

To evaluate the conversational search feature, we use the RAGAS framework (Es et al., 2024), focusing on the *Faithfulness* and the *Answer Relevance* of generated responses. Faithfulness evaluates if the generated answer is grounded in the given context, which is important to avoid hallucinations. Answer relevance evaluates if the generated answer actually addresses the provided question. We use GPT-4 to generate 50 random questions related to NLP, such as *"Define perplexity in the context of language models"*. Subsequently, we utilize GPT-3.5 (OpenAI, 2022) and GPT-4 in our conversational search pipeline described in §3.6 to generate grounded answers from retrieved publications. Finally, we use RAGAS to evaluate the generated responses. As shown in Table 5, both LLMs exhibit high faithfulness and answer relevance scores, indicating their ability to retrieve relevant publications from the RAG pipeline to effectively answer user queries based on provided contexts.

| Model | Faithfulness | Answer Relevance |
|---|---|---|
| gpt-3.5-turbo-0125 | 0.9661 | 0.8479 |
| gpt-4-0125-preview | 0.9714 | 0.8670 |

Table 5: Evaluation results of our conversational search pipeline. Metrics are scaled between 0 and 1, whereby the higher the score, the better the performance.

### 5.3 Comparison of Scholarly Literature Search Systems

We compare NLP-KG with other publicly accessible systems for scholarly literature search, including Google Scholar, Semantic Scholar, ORKG, NLP Explorer, and NLP Scholar. A feature comparison is shown in Table 6.

| | Google Scholar | Semantic Scholar | ORKG | NLP Explorer | NLP Scholar | NLP-KG |
|---|---|---|---|---|---|---|
| Keyword-based Search | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| NLP specific | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ |
| Fields of Study Tags | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ |
| Fields of Study Hierarchy | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ |
| Survey Filter | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ |
| Ask This Paper | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Conversational Search | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |

Table 6: Feature comparison of scholarly literature search systems.

The comparison shows that NLP-KG offers an extensive set of features providing users with a wide range of options to explore NLP research literature. Unlike popular systems such as Google Scholar and Semantic Scholar, NLP-KG is tailored specifically for NLP research, ensuring an accurate and efficient exploration experience. Moreover, NLP-KG is not limited to keyword-based searches, providing users with advanced search and retrieval features to explore the field of NLP.

## 6 Conclusion

This paper introduces NLP-KG, a system for search and exploration of NLP research literature. NLP-KG supports the exploration of unfamiliar fields by providing a high-quality knowledge graph of FoS in NLP and advanced retrieval features such as semantic search and filtering for survey papers. In addition, a LLM integration allows users to ask questions about the content of specific papers and unfamiliar concepts in NLP and provides answers based on knowledge found in scientific publications. Our model evaluations demonstrate strong classification and retrieval performances, making our system well-suited for literature exploration.

## Limitations

The construction of the FoS hierarchy graph depends on the personal choices of the domain experts, which may bias the final result. The hierarchy graph may not cover all possible FoS and offers potential for discussions as domain experts have inherently different opinions. As a countermeasure, we automatically extracted entities and relations from a corpus of NLP-specific documents and aligned the opinions of domain experts during the manual construction process.

We have limited the database of our system to papers published in the ACL Anthology and the arXiv cs.CL category. However, NLP research is also presented at other conferences such as AAAI, NeurIPS, ICLR, or ICML, which may not be included in our system.

## Ethical Considerations

NLP-KG supports the search and exploration of NLP research literature in unfamiliar fields. To enable an intuitive user experience, the application integrates LLM-based features. However, LLMs (e.g., GPT-4, used in this work) are computationally expensive and require significant compute resources. Additionally, although we aim to minimize model hallucinations by grounding the model responses in knowledge retrieved from scientific publications, the integrated LLM can nevertheless make mistakes. Therefore, users should always check important information provided by our LLM-based features.

## Acknowledgements

Many thanks to Matthias Aßenmacher for his much appreciated proofreading efforts. We also thank Nektarios Machner, Phillip Schneider, Stephen Meisenbacher, Mahdi Dhaini, Juraj Vladika, Oliver Wardas, Anum Afzal, and Wessel Poelman for helpful discussions and valuable feedback. In addition, we thank Ferdy Hadiwijaya, Patrick Kufner, Ronald Ernst, Furkan Yakal, Berkay Demirtaş, and Cansu Doğanay for their contributions during the implementation of our system. Finally, we thank the anonymous reviewers for their useful comments.

## References

Sören Auer, Allard Oelen, Muhammad Haris, Markus Stocker, Jennifer D'Souza, Kheir Eddine Farfar, Lars Vogt, Manuel Prinz, Vitalis Wiens, and Mohamad Yaser Jaradeh. 2020. Improving access to scientific literature with knowledge graphs. *Bibliothek Forschung und Praxis*, 44(3):516–529.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 758–759, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158, St. Julians, Malta. Association for Computational Linguistics.

Sergey Feldman. 2020. Building a better search engine for semantic scholar.

Mohamad Yaser Jaradeh, Allard Oelen, Kheir Eddine Farfar, Manuel Prinz, Jennifer D'Souza, Gábor Kismihók, Markus Stocker, and Sören Auer. 2019. Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge. In *Proceedings of the 10th International Conference on Knowledge Capture*, K-CAP '19, page 243–246, New York, NY, USA. Association for Computing Machinery.

Rodney Kinney, Chloe Anastasiades, Russell Authur, Iz Beltagy, Jonathan Bragg, Alexandra Buraczynski, Isabel Cachola, Stefan Candra, Yoganand Chandrasekhar, Arman Cohan, Miles Crawford, Doug Downey, Jason Dunkelberger, Oren Etzioni, Rob Evans, Sergey Feldman, Joseph Gorney, David Graham, Fangzhou Hu, Regan Huff, Daniel King, Sebastian Kohlmeier, Bailey Kuehl, Michael Langan, Daniel Lin, Haokun Liu, Kyle Lo, Jaron Lochner, Kelsey MacMillan, Tyler Murray, Chris Newell, Smita Rao, Shaurya Rohatgi, Paul Sayre, Zejiang

Shen, Amanpreet Singh, Luca Soldaini, Shivashankar Subramanian, Amber Tanaka, Alex D. Wade, Linda Wagner, Lucy Lu Wang, Chris Wilhelm, Caroline Wu, Jiangjiang Yang, Angele Zamarron, Madeleine Van Zuylen, and Daniel S. Weld. 2023. The semantic scholar open data platform.

Kyle Lo, Zejiang Shen, Benjamin Newman, Joseph Chang, Russell Authur, Erin Bransom, Stefan Candra, Yoganand Chandrasekhar, Regan Huff, Bailey Kuehl, Amanpreet Singh, Chris Wilhelm, Angele Zamarron, Marti A. Hearst, Daniel Weld, Doug Downey, and Luca Soldaini. 2023. PaperMage: A unified toolkit for processing, representing, and manipulating visually-rich scientific documents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 495–507, Singapore. 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.

Saif M. Mohammad. 2020. NLP scholar: An interactive visual explorer for natural language processing literature. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 232–255, Online. Association for Computational Linguistics.

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI*.

OpenAI. 2023. Gpt-4 technical report.

Malte Ostendorff, Nils Rethmeier, Isabelle Augenstein, Bela Gipp, and Georg Rehm. 2022. Neighborhood contrastive learning for scientific document representations with citation embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11670–11688, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Monarch Parmar, Naman Jain, Pranjali Jain, P. Jayakrishna Sahit, Soham Pachpande, Shruti Singh, and Mayank Singh. 2020. Nlpexplorer: Exploring the universe of nlp papers. In *Advances in Information Retrieval*, pages 476–480, Cham. Springer International Publishing.

Jason Priem, Heather Piwowar, and Richard Orr. 2022. Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts.

S. E. Robertson and S. Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, page 232–241, Berlin, Heidelberg. Springer-Verlag.

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.

Zhihong Shen, Hao Ma, and Kuansan Wang. 2018. A web-scale system for scientific knowledge exploration. In *Proceedings of ACL 2018, System Demonstrations*, pages 87–92, Melbourne, Australia. Association for Computational Linguistics.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2023. SciRepEval: A multi-format benchmark for scientific document representations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5548–5566, Singapore. Association for Computational Linguistics.

Mayank Singh, Pradeep Dogga, Sohan Patro, Dhiraj Barnwal, Ritam Dutt, Rajarshi Haldar, Pawan Goyal, and Animesh Mukherjee. 2018. CL scholar: The ACL Anthology knowledge graph miner. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 16–20, New Orleans, Louisiana. Association for Computational Linguistics.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, page 243–246, New York, NY, USA. Association for Computing Machinery.

Ayah Soufan, Ian Ruthven, and Leif Azzopardi. 2022. Searching the literature: An analysis of an exploratory search task. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, CHIIR '22, page 146–157, New York, NY, USA. Association for Computing Machinery.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: Extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, page 990–998, New York, NY, USA. Association for Computing Machinery.

Kuansan Wang, Zhihong Shen, Chiyuan Huang, Chieh-Han Wu, Yuxiao Dong, and Anshul Kanakia. 2020. Microsoft Academic Graph: When experts are not enough. *Quantitative Science Studies*, 1(1):396–413.

Benjamin Weitz and Ulrich Schäfer. 2012. A graphical citation browser for the ACL Anthology. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1718–1722, Istanbul, Turkey. European Language Resources Association (ELRA).

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zhicheng Dou, and Ji-Rong Wen. 2024. Large language models for information retrieval: A survey.