3rd NFDI4Energy Conference: Defining Energy Research Data Management Together

Proceedings to the 3rd NFDI4Energy Conference - Full Papers

https://doi.org/10.52825/ocp.v9i.3306



Published: 23 Mar. 2026

# Evaluating Vector-Based Search and Question Answering Approaches for an Information System

Michael Dembach[1,*] and Sophie Decher[1]

[1]Fraunhofer FKIE, Germany

*Correspondence: Michael Dembach, michael.dembach@fkie.fraunhofer.de

**Abstract.** This paper presents two experiments conducted within the context of an information system. First, it evaluates the potential of vector-based document retrieval in contrast to ontology-based query expansion using a manually curated categorization scheme that is employed in actual practice rather than being constructed specifically for the experiment. Second, it compares the output of a RAG system to that of a group of human domain experts. The findings reveal that a vector-based approach is more effective for this use case and that RAG-generated texts may be able to stylistically compete with those of experts though content needs to be checked.

**Keywords:** Ontology, RAG, Search, Question Answering, Information System, Explainability

## 1. Introduction

An effective research information system must ensure reliable access to scientific knowledge, which fundamentally depends on a well-designed search functionality. Beyond retrieval performance, explainability and transparency are central design principles: users derive substantial benefits from understanding how and from where information is obtained. Such transparency not only enhances usability but also fosters trust in the system's outputs.

This study consists of two experiments related to the process of information retrieval. Both experiments are conducted on data from the EnArgus information system. The EnArgus information system allows users to access research funding data as well as a Wiki of energy-related topics via an online portal. The goal of EnArgus is to make energy research funding more transparent and accessible to researchers, companies, funding institutions, ministries, and the general public. The current study investigates two search mechanisms that might improve system useability and performance. In a first experiment, we investigate two approaches to document ranking and compare them: 1) ontology-based query expansion and 2) vector-based semantic search. The two search systems are evaluated on a set of search queries and nDCG is used as metric. This first approach aims to evaluate:

**RQ1**: How do document rankings produced by a vector-based search system compare to those produced by a system driven by ontology-based query expansion?

The second experiment concerns itself with LLM-supported question answering based on RAG-retrieved context documents:

**RQ2**: Given a set of relevant documents as context, how does a RAG-generated answer compare to a human answer?

The remainder of this paper is organized as follows. Section 2 introduces the core technologies evaluated in this study, namely ontology-based search, vector search, and retrieval-augmented generation (RAG), and shortly reviews the seminal literature underlying these approaches. In Section 3, the methodology and findings for RQ1 are presented and discussed. The methodology and findings for RQ2 can be found in Section 4. A concluding discussion and future directions are presented in Section 5.

## 2. Literature Review

The semantic concept behind ontology augmented query expansion is that of set-theoretic formal semantics [1]. It serves as a way of increasing "lexical/semantic overlap between a user query and relevant documents" [2]. For example, a search string, e.g. *windmill*, can be expanded with related concepts from an ontology, e.g. *rotor*, which provides context for the original query and can help with word sense disambiguation [3], [4].

In vector space models (VSMs) of semantics, the context in which a word occurs shapes its meaning [1]. Since Mikolov [5] showed that semantic proximity can be learned from co-occurrence and subsequently expressed through metrics, and Reimers [6] expanded this approach to sentences and whole documents, it seems of interest to apply this approach to document retrieval tasks.

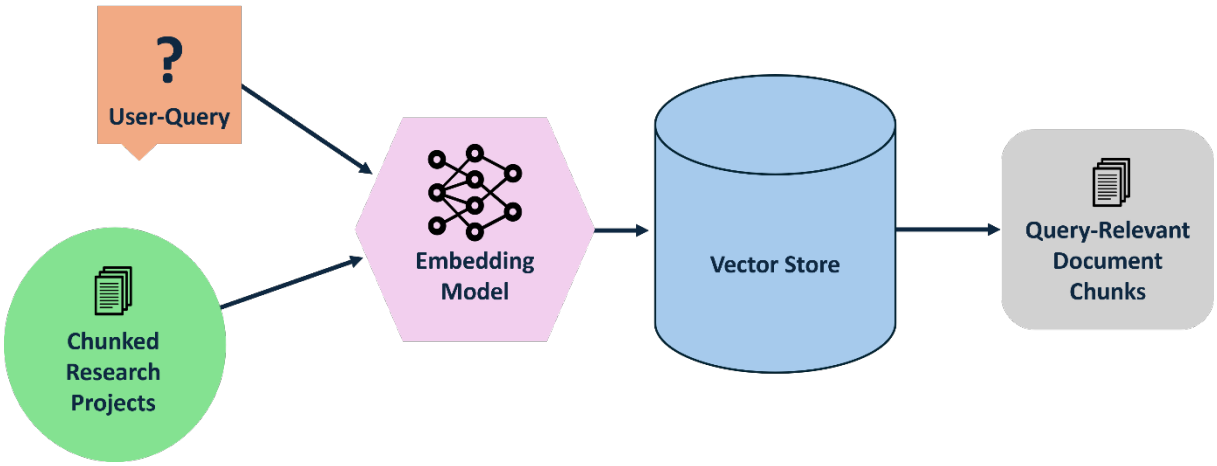


***Figure 1.*** *Vector-based search architecture*

Document retrieval constitutes the foundation of retrieval-augmented generation (RAG), an approach in which a large language model (LLM) is supplied with external documents to answer specific queries (cf. Figure 1). Most RAG systems rely on embedding-based retrieval, which in turn requires the query to be embedded into the same representation space [7].

## 3. Search Function Comparison

### 3.1 Building Corpora of Ranked Texts

The basis of this evaluation is a mapping of documents to sixteen thematic categories and four superordinate categories. This categorization was not originally created for the current study; rather, it reflects an internal schema originally devised for internal use during the EnArgus project. As such, it might be arbitrary to some degree—as every categorization ultimately is—but this top-down categorization serves as an objective, externally-defined starting point for the evaluation. As this schema is restricted from public use, variables will be used in place of actual category labels throughout this paper.

The categorical mapping of texts was used as ground truth for the experiment. The documents *d* are mapped to exactly one category *C* each, which is a subset of a superordinate category *SC*. We use the label of a category *C* as a search query, e.g. *power plants and solar energy.* A document can then be assigned one of three possible ranks (cf. Figure 2). Rank 1: The document is a member of the category ($d \in C$). Rank 2: The document is a member of the superordinate category but not of the category ($d \in CS \backslash C$). Rank 3: The document is neither a member of the category nor of the superordinate category ($d \notin CS$). A search algorithm can then be measured by how close it can replicate this categorization.

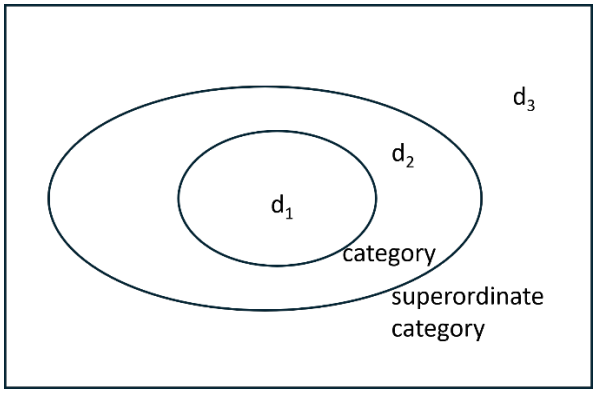


***Figure 2**. Ranking for a document based on its set membership in a taxonomy of categories.*

Sixteen subcorpora, one for each category, were created, each containing three documents from that same category (Rank 1), five documents of a different category with the same superordinate category (Rank 2) and seven documents of a different superordinate category (Rank 3). The evaluation described below determines how well two search algorithms were able to recreate these rankings.

## 3.2 Normalized Discounted Cumulative Gain (nDCG)

nDCG is a commonly used metric for evaluating the performance of ranking algorithms. Given relevance judgments for documents with respect to a query, an ideal ranking can be constructed, which serves as the reference ranking.

For each document *d* at rank position *i*, a graded relevance value $rel_i$ must be assigned. For this experiment, this value was constructed heuristically by reversing the ranking:

$$rel_i = \max(rank) - i + 1 \tag{1}$$

In order to rank a set of documents found by the search algorithm without regard for the position of a single document in the ranking, cumulative gain (CG) can be used, which sums up all the graded relevancies for a set of *n* documents:

$$CG_n = \sum_{i=1}^{n} rel_i \tag{2}$$

For most evaluation scenarios, including the one of this paper, it is more useful to take the document order into account by penalizing relevant documents that appear lower in the ranking. This is called discounted cumulative gain:

$$DCG_n = \sum_{i=1}^{n} \frac{2^{rel_i}-1}{log_2(i+1)} \tag{3}$$

For different sets of documents, the DCG cannot be compared, as set length may vary. To gain a normalized value, the DCG is divided by the ideal DCG, i.e. all documents ranked by their relevance. This is referred to as the nDCG:

$$nDCG_n = \frac{DCG_n}{IDCG_n} \quad (4)$$

The value of the nDCG is between zero and one.

## 3.3 Results

A first evaluation investigated the ranking accuracy using only query tokenization. Table 1 shows that this method already achieves acceptable results, most likely due to the fact that the documents used contain semantically relevant keywords. It also suggests, however, that the categories are semantically sound.

***Table 1.*** *NDCG values*

| **Query** | **nDCG@15** Tokens Only | **nDCG@15** Query Expansion | **nDCG@15** Vector Based |
|---|---|---|---|
| Query_1 | 0,833530654 | 0,872363 | 0,991814 |
| Query_2 | 0,835528812 | 0,850485 | 0,932562 |
| Query_3 | 0,772786072 | 0,850485 | 0,90781 |
| Query_4 | 0,954409091 | 0,954409 | 0,91538 |
| Query_5 | 0,919700094 | 0,9197 | 0,973257 |
| Query_6 | 0,806016937 | 0,850485 | 0,866895 |
| Query_7 | 0,977549003 | 0,977549 | 0,992816 |
| Query_8 | 0,935685279 | 0,850485 | 0,885592 |
| Query_9 | 0,850484602 | 0,850485 | 0,951133 |
| Query_10 | 0,88614361 | 0,850485 | 0,902931 |
| Query_11 | 0,884802135 | 0,850485 | 0,946316 |
| Query_12 | 0,977549003 | 0,977549 | 0,915733 |
| Query_13 | 0,850484602 | 0,850485 | 0,941379 |
| Query_14 | 0,830969197 | 0,840745 | 0,864211 |
| Query_15 | 0,799388273 | 0,883791 | 0,957223 |
| Query_16 | 0,815445793 | 0,850485 | 0,982279 |
| **Sum** | **13,93047316** | **14,08047** | **14,92733** |

The query expansion approach discussed in the current paper requires the inputted search query to first be tokenized, as was also done in Section 4.1. In a second step, the search query tokens were mapped to class labels in the ontology in order to calculate the token overlap between the search query and ontology. For each mapped ontology class label, any synonyms and subclasses were then also added. The resulting overlapping terms were assigned weights: one-to-one matches to ontology class labels were weighted as 1.0; synonym matches were assigned 0.9; and subclass matches were assigned 0.7. Lastly, term weights were aggregated to create a single score for each search query.

The query expansion shows better results for nearly all subcorpora and an overall better performance of ≈0.14. This may appear to be a small improvement, but considering how short the documents are—leaving limited space for overlap between the query, ontology labels and the document's text—this still shows that query expansion is a valid method.

For the vector-based search function, the documents were embedded with the pre-trained `nomic-embed-text` embeddings model [8], [9]. No chunking was needed as all documents were fewer than 100 tokens in length. Each search query was vectorized with the same embeddings model and compared to the vectors of the embedded documents. The documents that were semantically similar to the query (determined via cosine similarity) were then retrieved ($k$=15).

The vector-based approach shows significantly better results, outperforming the query expansion method presented in Section 4.2 by a margin of 0.84686. The fact that the query strings were only phrases and not whole sentences also speaks in favor of this approach. It is likely that the vector-based approach would outperform query expansion even more if the queries were whole sentences, as the multi-dimensional vectors can represent the relations between different words in detail.

### 3.4 Discussion

The results show that the vector-based approach performs more accurately when it comes to ranking documents for this use case, while the ontology-based query expansion still produces reasonable results. It must be taken into consideration that the documents were quite short, which may have caused difficulties for the query expansion approach. It has also to be noted, however, that the queries were only phrases and not full sentences—the vector-based approach might be still more effective on full sentence queries. Query expansion might still be the better option for some use cases, however, especially when it comes to increasing explainability, as the process of document ranking can be made more transparent.

## 4. RAG-Powered Question Answering

### 4.1 Methodology

#### 4.1.2 Data

For this experiment, a group of eight domain experts was asked to create a list of possible questions that users might search for. The experts were then prompted to write short answers to the questions (*How would you want an LLM to answer this question in the context of our system?*). If it was clear that a question would not be answerable with the data available to the system, or was otherwise outside the scope of the project, answers were permitted to include phrases such as *I'm unable to answer this question.* This resulted in a gold standard dataset of fifteen question-answer pairs.

```
    Deine Aufgabe ist es, Fragen zur Energieforschung sachlich und
nach wissenschaftlichen Standards zu beantworten. Formuliere eine
prägnante und vollständige Antwort auf die folgende Frage und ver-
wende die bereitgestellten Kontextdokumente, um deine Antwort zu
stützen.

    Kontextdokumente:
```

***Figure 3**. System prompt for answer generation*

Using the fifteen gold standard questions, we then prompted three LLMs to generate answers. Each model was shown the same system prompt (cf. Figure 3) and a selection of predetermined semantically relevant texts from the project data as context. This was done to emulate the expert knowledge that a human might have. We chose to compare three midsized multilingual open-source models: `deepseek-r1:32b`[1], `mistral-small 3.2:24b`[2] and `qwen 3:32b`[3]. The resulting corpus contained fifteen questions, each with four possible responses (totalling nineteen responses).

[1] https://ollama.com/library/deepseek-r1:32b
[2] https://ollama.com/library/mistral-small3.2:24b
[3] https://ollama.com/library/qwen3:32b

### 4.1.3 Evaluation

A group of nine participants familiar with our information system were given a pairwise comparison task in order to evaluate which of the four response categories (human expert, deepseek-r1:32b, mistral-small 3.2:24b and qwen 3:32b) produced the best results. The corpus created in Section 5.1.2 was divided into three different randomized sets of thirty questions, each accompanied by a pair of responses. Questions could and did appear more than once in each set. Each of these three sets was evaluated by three participants. Participants were told to imagine that the questions presented to them were being answered by a chatbot powered by data from our project. They were asked to select the better of the two presented answers (*Welcher Text ist eine sachlich korrektere und präzisere Antwort auf die Suchanfrage?).*

## 4.2 Results

Table 2 shows the resulting ranking from the pairwise comparison task. The responses generated by `mistral-small 3.2:24b` were selected most often by participants (*n*=82), followed by `qwen 3.32:24b` (*n*=80). Responses written by human experts were ranked in third place (*n*=62).

***Table 2.*** *Response category ranking.*

| Ranking | Response category | Wins |
|---|---|---|
| 1 | Mistral-small 3.2:24b | 82 |
| 2 | Qwen 3:32b | 80 |
| 3 | Human expert | 62 |
| 4 | Deepseek-r1:32b | 46 |

Human expert responses were most often selected when paired with responses from `deepseek-r1:32b` (*n*= 27; cf. Figure 4), the least successful category. When paired with responses from mistral-small 3.2:24b or qwen 3:32b, human expert responses were selected less often than the responses of the two LLMs. All nine participants selected responses from all four response categories (cf. Figure 5). Dividing the results by question reveals that certain response categories were repeatedly selected for particular questions (cf. Figure 6). Questions 3, 4, and 6 were best answered by `mistral-small 3-2:24b`, while questions 9, 13, and 15 were best answered by `qwen 3:32b` and question 10 was best answered by `deepseek-r1:32b`. The responses written by human experts were not disproportionately selected for any particular question.

## 4.3 Discussion

During the creation of the corpus for this experiment, all three LLMs had difficulties with longer contexts; shortening the context solved this problem for deepseek-r1:32b and mistral-small 3.2:24b, but qwen 3:32b continued to have problems. Deepseek-r1:32b also often inserted Chinese and Japanese characters during generation, though the context documents were all written in German. Participants found this confusing. This makes it all the more remarkable that in 45 of the deepseek-r132b/human expert answer pairs, participants still selected texts generated by deepseek-r132b18 times (cf. Figure 4).

With regard to our second research question (RQ2), it is thus apparent that human expert answers do not clearly outperform LLM-generated answers. RAG-generated answers are able to compete with and potentially even be seen as on par with human-written texts.

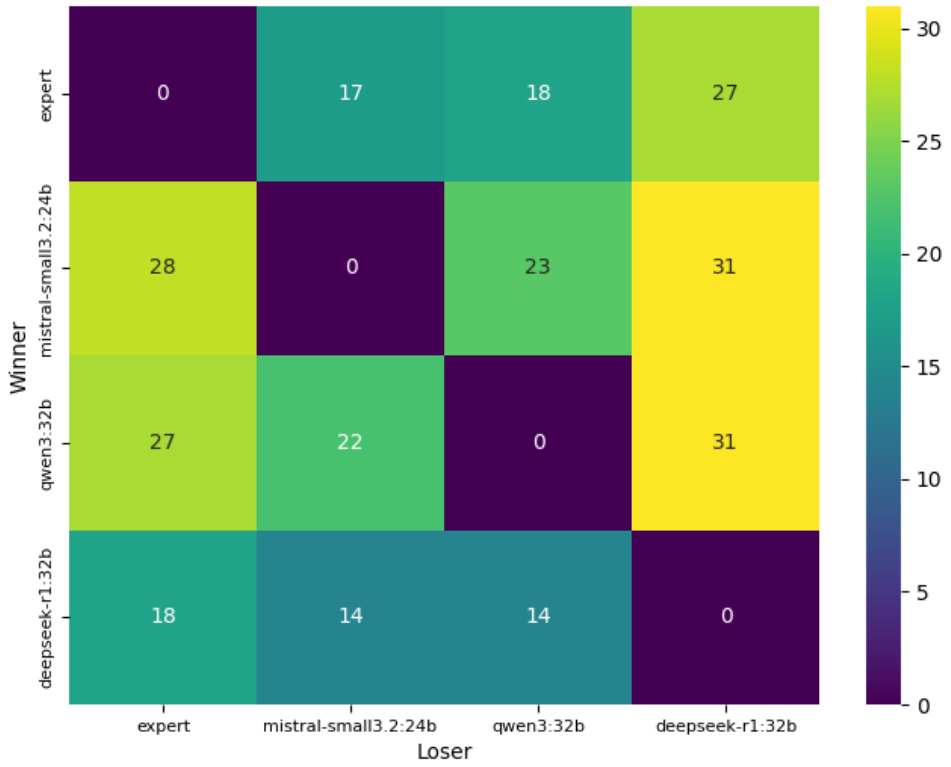


***Figure 4.*** *Win-loss matrix of the four response categories.*

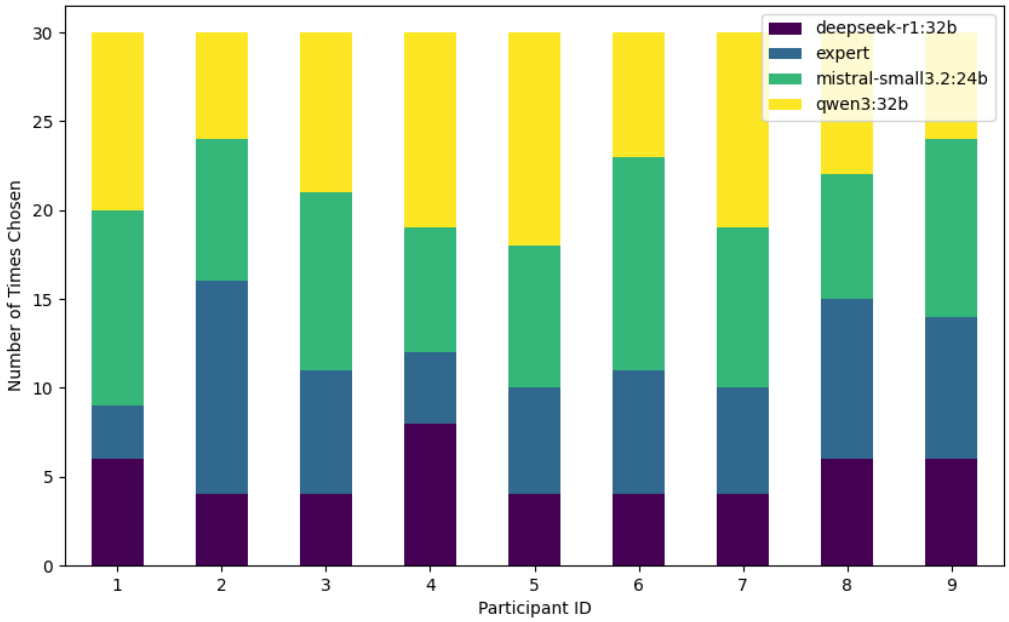


***Figure 5.*** *Response choices by participant.*

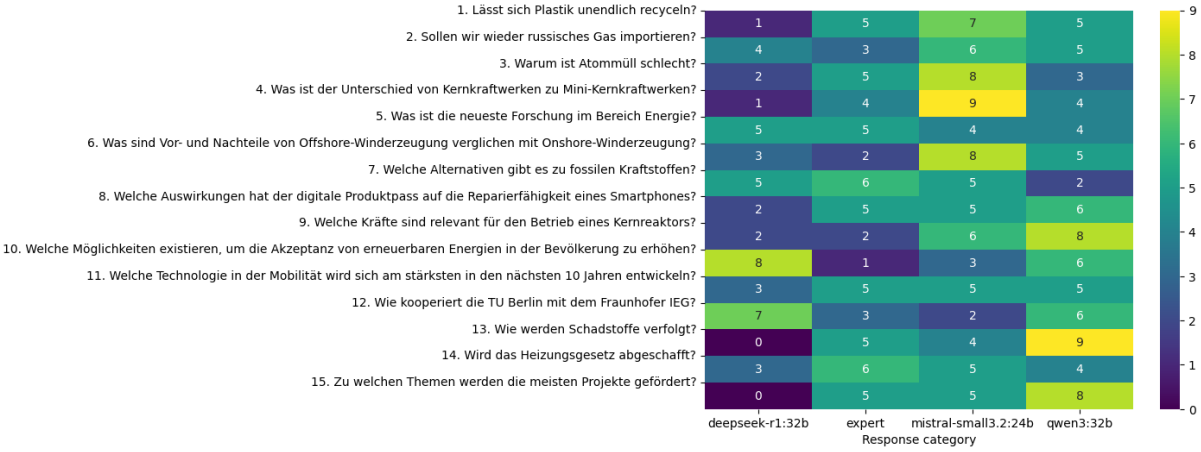


***Figure 6.*** *Response choices by participant.*

## 5. Conclusion

Both document ranking and question answering are important tasks that need to be optimized in order to create an efficient and user-friendly information system. The first experiment described in Section 3 investigates how well information can be retrieved. The findings indicate that a vector-based approach is best able to recreate the ranking of the original dataset (RQ1). It is likely that those findings can be generalized to other use cases. One disadvantage to using a vector-based approach is that such LLM-based approaches may struggle with dynamic domains where new terms are constantly emerging, as is the case in the domain of the use case at hand [8]. Query expansion also produced reasonable results in this experiment and has the added benefit of increased explainability. It is also easier to control which information is in the ontology and to update it accordingly if new domain knowledge needs to be added. There are also other possible ways of leveraging the information in an ontology, such as with ontology embeddings, a possible future direction [10], [11].

Meanwhile, the second experiment in Section 4 evaluates how well an LLM can generate answers to energy related questions, given a set of retrieved documents as context. These generated texts are then compared and manually evaluated by domain experts. Surprisingly, the texts written by human experts were not always preferred. Out of the four response categories, texts generated by `mistral-small 3-2:24b` and `qwen 3:32b` were most often selected (RQ2). on retrieved documents.

The experiments described above are not without limitations. The documents used in Experiment 1 are short in length, which likely decreased the task difficulty for the algorithms we compared. The dataset's value, however, lies in that it was not synthetically designed but has a real-life application, making this evaluation more realistic. Experiment 2 was limited in the number and size of the LLMs that we were able to compare, as domain expert participants were difficult to source. A larger gold standard set of search queries and human-generated answers would also allow for more accurate evaluation of the RAG approach.

## Data availability statement

The ontology used is the EnArgus ontology (https://www.enargus.de/enargus.html). The documents that were embedded and used for the query evaluations are restricted. The corpus used in Section 5 will be uploaded to fordatis (https://fordatis.fraunhofer.de).

## Funding

This work is funded by Projektträger Jüllich

## Author contributions

The authors contributions are as follows:

First Author: Project administration, Conceptualization, Data curation, Formal analysis, Writing – original draft

Second Author: Conceptualization, Data curation, Formal analysis, Writing – original draft, Writing – review & editing

## Competing interests

The authors declare that they have no competing interests.

## References

[1] S. Clark, “Vector Space Models of Lexical Meaning,” The Handbook of Contemporary Semantic Theory. S. Lapind and C. Fox (eds), 2015, Chapter 16, doi: https://doi.org/10.1002/9781118882139.ch16

[2] J. Bhogal, A. Macfarlane, P. Smith, “A Review of Ontology Based Query Expansion,” Information Processing & Management, 43, 4, pp. 866–886, Jul. 2007, doi: https://doi.org/10.1016/j.ipm.2006.09.003

[3] A. Broder, “A Taxonomy of Web Search,” ACM Sigir Forum, 36, 2, pp. 3–10, Sept. 2002, doi: https://doi.org/10.1145/792550.792552

[4] T. Mikolov et al., “Efficient Estimation of Word Representations in Vector Space”. in Advances in Neural Information Processing Systems, *arXiv:1310.4546*, 2013, doi: https://doi.org/10.48550/arXiv.1301.3781

[5] N. Reimers et al. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In Conference on Empirical Methods in Natural Language Processing 2019, doi: https://doi.org/10.48550/arXiv.1908.10084

[6] M. Douze et al., “The Faiss Library.” arXiv:2401.08281v4, pp. 1–25, Oct. 2025, doi: https://doi.org/10.48550/arXiv.2401.08281

[7] P. Lewis et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks” arXiv:2005.11401, May 2021, doi: https://doi.org/10.48550/arXiv.2005.11401

[8] M. Li, X. Lv, J. Zou, T. Chen, C. Zhang, S. An, E. Nie, G. Zhou, “Query Expansion in the Age of Pre-Trained and Large Language Models: A Comprehensive Survey.” arXiv:2509.07794v2, pp.1–36, Oct. 2025, doi: https://doi.org/10.48550/arXiv.2509.07794

[9] Z. Nussbaum, J. X. Morris, B. Duderstadt, A. Mulyar, „Nomic Embed: Training a Reproducible Long Context Text Embedder,” Transactions on Machine Learning Research Feb 2025, doi: https://doi.org/10.48550/arXiv.2402.01613

[10] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Wei Chen, Tie-Yan Liu. A Theoretical Analysis of Normalized Discounted Cumulative Gain (NDCG) Ranking Measures. In Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013), doi: https://doi.org/10.48550/arXiv.1304.6480

[11] J. Chen, O. Mashakova, F. Zhapa-Camacho, R. Hoehndorf, Y. He, I. Horrocks, “Ontology Embedding: A Survey of Methods, Applications and Resources, IEEE Transactions on Knowledge and Data Engineering, 37, 7, pp. 4193–4212, Jul. 2025, doi: https://doi.org/10.1109/TKDE.2025.3559023