Continuous Online Evaluation of Recommendation Strategies in Social Science Academic Search

Mehmet Deniz Türkmen[0009−0003−8705−7770] ✉
Daniel Hienert[0000−0002−2388−4609]

GESIS - Leibniz Institute for the Social Sciences, Cologne, Germany
✉deniz.tuerkmen@gesis.org, daniel.hienert@gesis.org

Abstract. Delivering relevant recommendations in academic search engines is a complex task due to the diversity of subject areas, information types, and user preferences. In this case study, we address these challenges by integrating and evaluating a range of recommendation systems within GESIS Search – a domain-specific search engine for the social sciences that provides researchers with access to research data, publications, variables, and measurement instruments. To support continuous, real-time evaluation of multiple recommendation strategies with actual platform users, we utilize the STELLA evaluation framework. We implement and compare a diverse set of algorithms, including traditional lexical similarity, semantic document similarity by using transformer-based embeddings, and session-based recommendations based on click paths from historical user sessions. Our results show that users prefer recommendations based on semantic similarity, which outperformed term-similarity and session-based methods. However, the performance of recommenders varies across categories within GESIS Search, suggesting that information-seeking behavior differs by information type. Overall, our study provides insights into how continuous evaluation can be incorporated to develop recommendations that better align with the preferences in academic search portals.

Keywords: Continuous Evaluation, Recommendations, Academic Search, Social Sciences

1 Introduction

Besides large cross-disciplinary search portals for publications [34, 53], there are numerous academic search engines that focus on specific domains, such as biomedicine [80], natural sciences [24], humanities [70, 76], economics [17, 86], and computer science [46, 81]. These platforms provide access to subject-specific information, such as publications, as well as other types of content, such as research data [11, 20], images, videos & audio [76], and research output & software [20, 26]. Users of these portals are often scientists who rely on the information they find as a basis for their own research [50, p. 5]. Consequently, the topics they search for tend to be highly domain-specific and vary substantially across disciplines [23].

In addition to ad hoc search, recommendations provide users with suggestions tailored to their information needs [73]. However, the relevance of recommendations depends on various factors, such as the domain, subject area, type of information, search topics, and the portal's user group. In commercial systems, recommendations are often evaluated directly with the user base, for example, prominently through A/B testing [57]. This allows users to implicitly indicate which recommendations are more relevant through their interactions. In contrast, online evaluation is not yet widely used in academic search engines and is mostly limited to publications [59]. Recommendations for other information types, such as research data, have been explored sparingly so far, even though recommendations from other researchers seem to play an important role in dataset search [43].

In this case study, we evaluate different types of recommendation strategies in an academic search engine for social science information. To enable continuous online experimentation, we employ the STELLA evaluation framework [10], which allows us to compare multiple recommendation systems through interleaving. In this setup, recommendations generated by different systems are combined into a single ranked list, and user interactions are used to infer system preferences. This approach enables continuous deployment and evaluation of new recommendation methods without significantly affecting user experience.

We evaluate various recommendation approaches based on lexical similarity, semantic similarity, and historical user sessions. Additionally, we perform a category-specific evaluation to better understand how recommender performance varies across different types of scholarly content. Our findings show that semantic similarity–based recommenders achieve the strongest overall performance in our use case, outperforming both lexical and session-based methods. However, we also observe that the performance can vary between categories within GESIS Search. In other words, certain recommendation strategies perform better for specific information types, suggesting that users' information needs vary with the nature of the content. These findings highlight the importance of continuous online evaluation for developing effective recommendation systems in academic search environments across diverse domains and use cases.

2 Related Work

2.1 Recommendations in Academic Search

Recommender systems have been studied in the field of digital libraries for a long time [25, 73]. Recommendations in academic search typically focus on related-article suggestions [5, 27], but have also been applied to venue recommendations [1, 18, 19, 84], personalized research feeds [48], related patent identification [44, 65], and scientific video recommendations [56]. Existing approaches range from content-based techniques to methods that exploit citation networks. In systems with sufficient interaction data, collaborative filtering [31] and hybrid methods combining textual, citation, and behavioral signals [74] have also been explored.

Content-based approaches typically rely on textual similarity between documents by using representations derived from titles, abstracts, keywords or full texts. Earlier approaches primarily employed TF–IDF-based representations [32], while more recent systems use neural embeddings [41] and transformer-based representations [29] to capture semantic similarity between documents more effectively. Another important line of research focuses on citation-based recommendation methods, which exploit bibliographic networks to identify related work [39, 45, 78]. Such approaches are particularly effective in scholarly domains, where citations provide a strong signal of topical and methodological similarities. Similarly, several studies utilize Knowledge Graphs (KGs) to improve recommendations [9, 82].

Compared to commercial platforms, personalization in academic search systems remains relatively limited, largely due to privacy concerns and the limited availability of persistent user profiles. Nevertheless, several studies have explored personalized recommendations based on user profiles obtained from publication history, reading behavior, or declared research interests [37, 49, 60].

In practice, recommendation components in academic search engines are often integrated directly into search result pages or document detail views, where they complement traditional ranked retrieval. Recent work has also explored conversational and generative interfaces that combine retrieval and recommendation to guide literature exploration more interactively [2, 55, 79].

Despite numerous works on academic recommendation systems, evaluation in this domain is largely limited to offline methods [5, 16, 85]. On the other hand, several studies argued that test collections fail to reflect real-world user behavior and evolving information needs, motivating a shift toward online evaluation [4,6]. Accordingly, Beel et al. [3] introduced Mr.DLib, a living-lab evaluation framework designed specifically for scholarly recommendation settings. Other studies have further examined how users interact with recommendations and the factors that affect that, such as demographics [8], persistence of displayed results [7], and material type [14].

2.2 Continuous Evaluation in Information Retrieval

Evaluation in information retrieval has traditionally relied on the Cranfield paradigm, where systems are evaluated offline using static document collections, predefined topics, and manually created relevance judgments [15, 77]. Introduced by shared task campaigns such as TREC [28] and CLEF [61], this approach enabled reproducible and standardized comparison between IR systems. However, prior research has highlighted several limitations of static test collections, particularly their inability to reflect evolving document corpora, changing user needs, and advances in retrieval methods [35, 38, 66].

To address these limitations, researchers have explored evaluation approaches that are integrated into live systems. In this context, A/B testing has become a standard method for comparing IR systems based on user interaction signals [40]. Interleaving techniques have further improved sensitivity by combining results from different models into a single list and inferring user preferences from click

behavior [36, 62]. Together, these approaches have paved the way for continuous system assessment under real usage conditions.

In addition to industrial applications, the living lab paradigm has also been adopted in academic domains through shared-task campaigns that allow experimental systems to be deployed on operational platforms [68, 69]. Such environments provide opportunities to evaluate systems directly with real users while maintaining operational search services.

Overall, this body of work reflects a shift from static, one-time assessment toward iterative, deployment-oriented evaluation. Rather than replacing traditional test collection–based evaluation, continuous evaluation complements it by enabling ongoing system assessment in dynamic and real-world environments.

3 Evaluation and Search Systems

In this work, we deployed the STELLA evaluation framework[1] in the social science search platform GESIS Search[2]. In the following, we outline the principal functionalities of these systems.

3.1 STELLA Evaluation Framework

STELLA [10] is an open-source evaluation framework that enables portal operators and researchers to conduct experiments with real users in real-world environments. It provides an Evaluation-as-a-Service platform for living lab experiments involving ranking and recommender systems. Unlike traditional Cranfield-style evaluation approaches that rely on offline test collections, STELLA facilitates the assessment of experimental systems through user interactions and feedback. In addition to conventional A/B testing, the framework supports interleaving-based system comparison, where the outputs of two ranking or recommendation models are combined into a single result list for evaluation. STELLA's infrastructure consists of four main components:

Experimental Systems: A set of search or recommender systems to be evaluated within the framework.
Site: Platform where experimental systems interact with real users.
STELLA Server: The administrative component of the infrastructure, responsible for system management, dashboard access, storage of user interaction logs, and visualization of evaluation results.
STELLA App: A deployable application installed on participating platforms to conduct retrieval and recommender system experiments using real user interactions. It acts as a middleware between the host search engine (e.g., GESIS Search) and the external experimental systems. The app is mainly responsible for receiving rankings or recommendations from experimental systems, preparing the result list in line with the evaluation technique (i.e., A/B testing or

[1] https://github.com/stella-project
[2] https://search.gesis.org

interleaving), and conveying the final combined result to the platform. It also collects user interaction data such as clicks and views, which is subsequently transmitted to the STELLA Server for performance analysis. It ensures that experimental algorithms can be evaluated without disrupting the host platform by abstracting the evaluation logic from the main search portal.

3.2 Social Science Academic Search

GESIS Search [30] is an academic search platform that enables researchers to access and explore a broad range of social science resources within a single system. The platform supports cross-category search over research datasets, publications, survey variables, questionnaire items, survey instruments and tools, as well as webpages and publications from the GESIS library. In addition to keyword-based retrieval, the platform provides semantic links between information objects, allowing users to explore relationships between resources. For example, users can identify publications that cite specific research datasets or list variables associated with a particular study. To help researchers perform targeted searches, GESIS Search offers advanced filtering based on study collections and metadata attributes. The platform also includes functions for downloading datasets and related materials, which facilitate data reuse. Overall, the platform contains six categories of information. In this case study, recommendations are implemented for all categories except GESIS webpages.

Publications: ∼260,000 publications from open-access sources and literature linked to research datasets.
Research Data: ∼7,800 quantitative social science datasets covering Germany and Europe from 1945 to 2026, provided through the GESIS Data Archive.
Variables: ∼1.4 million survey variables extracted from questionnaires. Researchers can access high-quality variables with question texts, answer categories, and frequency tables.
Instrument & Tools: ∼620 resources related to survey design and data collection, such as pretested questionnaires, response scales, syntax files for analysis in statistical software, and guidelines for survey and behavioral data collection.
GESIS Library: ∼122,000 publications from the GESIS library, with a focus on empirical social science and applied computer science.
GESIS Webpages: ∼2,700 webpages from the GESIS website, including training materials, consulting services, and institutional information for social science researchers.

4 Experimental Recommender Systems

In this section, we describe the recommender systems implemented and evaluated within the online living lab environment.

4.1 Term Similarity

All documents in the Gesis Search index follow a shared metadata schema consisting of common fields such as title and abstract, complemented by type-specific metadata standards (e.g., DDI metadata [3] for research datasets). The term similarity recommender identifies related documents based on lexical overlap between these fields and those of a given source document. To implement this approach, we rely on the "more like this" (MLT) query in Elasticsearch [4]. The MLT query extracts text from the source document based on the field mapping. It then selects the top-k terms with the highest TF-IDF scores. These terms are used in a disjunctive query to retrieve and recommend the most similar documents from the index.

Listing 1 shows the instantiated MLT query. The "fields" parameter serves three purposes: (1) It determines which fields are used for text extraction from the source document and for querying candidate documents. (2) It allows weighting of fields; for example, terms appearing in titles can receive higher weights than terms in other fields. (3) It links fields to specific analyzers defined in the index mapping. For example, the field `title.partial` applies lowercasing and n-gram tokenization with token lengths between four and eight characters. This allows the query to find partial matches and retrieval of related terms such as "migration", "migrant", or "emigration". Overall, the field configuration provides detailed control over term extraction, field weights, and linguistic analysis.

Listing 1 The more-like-this query used for querying recommendations with similar terms

```
"query": {
  "more_like_this": {
    "fields": [
      "_all","title^10","topic^7","abstract^3","title_en^10","topic_en^7","abstract_en^3",
      "title.partial^0.4","topic.partial^0.3","abstract.partial^0.2","content.partial^0.4",
      "full_text^0.1"
    ],
    "like": [
      {
        "_index": index_name,
        "_id": item_id
      }
    ]
  }
}
```

4.2 Semantic Similarity

To explore beyond term-matching, we implemented recommenders based on semantic similarity using text embeddings. Unlike the term-similarity approach

[3] https://ddialliance.org/

[4] https://www.elastic.co/docs/reference/query-languages/query-dsl/query-dsl-mlt-query

relying on lexical overlap, this technique captures the meaning of the text by representing documents as dense vectors. To implement semantic recommendations, we first extract and concatenate textual information from selected metadata fields, such as abstract and title (Table 1), to form a textual representation for each scholarly item. Then, these are encoded into dense vectors using various embedding models.

In this study, we experimented with three embedding models: (1) *paraphrase-multilingual-mpnet-base-v2* [64], (2) *nomic-embed-text-v2-moe* [58], and (3) *all-MiniLM-L6-v2*[5]. These models were selected because they are multilingual and support the document corpus with a mixture of German and English documents and field contents. The first two models produce embedding vectors of length 768, whereas the third produces vectors of length 384.

The resulting vectors are indexed in Elasticsearch as dense vectors. During retrieval, we use Elasticsearch's K-nearest neighbor (KNN) query to identify semantically similar items based on cosine similarity between embedding vectors. Candidate documents with cosine similarity scores above 0.8 are retained as recommendations.

Information Category	**Fields**
Research Data	title, abstract, content_description
Variables	title, question_text
Instruments & Tools	title, abstract, topic, theory
Publications	title, abstract, topic
GESIS Library	title, abstract, topic

Table 1: Fields per information category used to build textual features for embeddings

Listing 2 shows an example KNN query. In the query, the "field" parameter is the field name where the embedding vector is stored in the ES index, while "query_vector" is the embedding vector itself of the target item. "k" represents the number of similar items to be found. "num_candidates" is the number of candidates to be explored before selecting the top-k items. Higher values improve retrieval quality at the cost of increased computational overhead. In our implementation, we set this value to 100 to balance efficiency and effectiveness. In addition, retrieval is restricted to items belonging to the same information category as the source item.

4.3 Session-based Click-Path Recommender

We implemented a session-based recommender that predicts users' next interaction based on anonymous click-path sequences in historical data. Unlike collaborative filtering approaches that require persistent user profiles, this method

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

Listing 2 The knn query used for retrieving based on embedding similarity

```
"knn": {
    "field": embedding_field,
    "query_vector": source_item_embedding,
    "k": k,
    "num_candidates": 100,
    "filter": {"term": {"type": source_item_type}}
}
```

relies on interactions within the current session and therefore supports anonymous recommendations. The model was trained on user interaction logs that span eight years, comprising 16M user interactions. Each session is represented as an ordered sequence of item interactions:

$$s = (i_1, i_2, \ldots, i_T)$$

where i_t denotes the item interacted at position t in the session and T is the session length. In our dataset, user sessions exhibit an average sequence length of $T = 5.4$ interactions with a standard deviation of 1.2.

The interaction sequence includes several types of user actions: (1) opening the detailed view of an item, (2) export actions such as exporting a citation, (3) dataset and questionnaire downloads, and (4) bookmarking items to the watchlist. Each of these actions indicates a specific user interest in an item. Every time the user changes their query, the click-path sequence is reset.

Given the subsequence $(i_1, i_2, ..., i_t)$, the task is to predict the next click i_{t+1}. Each item is represented with an embedding vector $e(i)$ of 768 length, which is generated using *nomic-embed-text-v2-moe* model [58]. To model sequential behavior, we employ a Recurrent Neural Network (RNN) that processes item embeddings in chronological order. Specifically, we use a single Gated Recurrent Unit (GRU) layer with 1000 hidden units to encode the current session's content:

$$h_t = GRU(h_{t-1}), e(i_t).$$

The final hidden state (h_t) of the GRU is then passed through a fully connected layer with 512 units to predict the embedding of the next likely click.

$$\hat{e}_{t+1} = FC(h_t).$$

The model is trained by minimizing cosine similarity loss between the predicted embedding and the embedding of the true next-clicked item:

$$L = 1 - cos(\hat{e}_{t+1}, e_{t+1}).$$

To expand the size of the training data and to improve robustness for short sessions, we apply prefix augmentation. In particular, for each original session $s = (i_1, i_2, ..., i_T)$, all prefixes

$$(i_1), (i_1, i_2), ..., (i_1, ..., i_{T-1})$$

are treated as independent training samples with their corresponding next-item targets $i_2, i_3, ..., i_T$.

At inference time, recommendations are generated through a KNN search in the embedding space using cosine similarity between the predicted embedding and candidate item embeddings. The top-k items with the highest similarity are returned as recommendations.

5 Experiments

In this section, we describe the evaluation use case, experimental methods used, and the results.

5.1 Use Case: Item Recommendation in GESIS Search

Use Case Overview: We focus on developing and evaluating recommendation systems across five distinct categories in GESIS Search: Research Data, Variables, Instruments & Tools, Publications, and GESIS Library. The recommendation section is integrated into the detailed view of each record. It displays up to 10 items in the same category as the currently viewed item; for instance, a user viewing a research dataset will see a list of similar datasets, as shown in Figure 1.

Evaluation: To determine the best approach for our use case, we implemented five recommender systems representing three distinct algorithmic classes. These systems were deployed as Docker containers and registered within the STELLA framework. The evaluation is based on the interleaving method. When a user views an item, the STELLA App concurrently requests recommendations from two candidate systems, merges their outputs into a single interleaved list, which is displayed to the user. User clicks on recommended items are recorded and attributed to the respective recommendation system. After a defined time period, the system with more clicks is considered the winner of the comparison.

User Interface and Interaction: Recommendations are displayed in the recommendations section as a standard hit list, allowing users to assess relevance based on titles, authors, and abstracts. Additional actions, including data access, full-text links, and citation export functions, are displayed alongside the recommendations. Although the exact layout and the specific metadata displayed vary slightly by item type (e.g., variables vs. publications), the core functionality is consistent. Clicking on a title opens the full view, where all metadata and available functions can be explored. Overall, this interface provides a natural context for users to interact with recommendations during exploratory search [30].

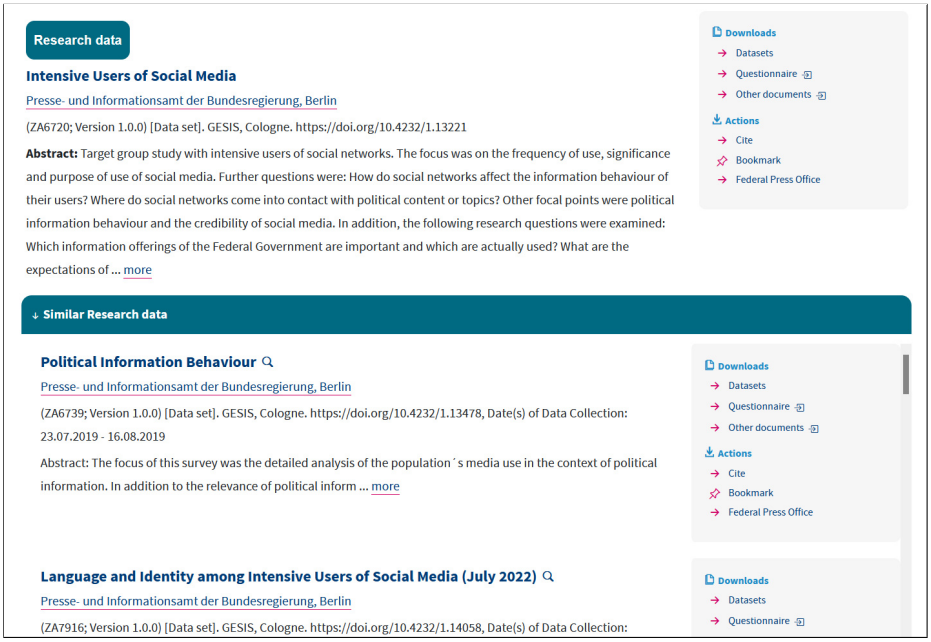

Fig. 1: Recommendations in the section *Similar Research Data* for a given research dataset about social media usage

5.2 Experimental Method

We evaluated five recommender systems using interleaving on the GESIS Search platform. Interleaving enables sensitive pairwise comparisons by combining the output of two systems into a single result list. Accordingly, we compared each pair of systems, resulting in a complete set of pairwise evaluations across all systems.

For all comparisons, we used team-draft interleaving (TDI) [63], which creates a single result list by selecting items from the two systems in a randomized draft order, minimizing bias. User interactions with the interleaved rankings were interpreted as implicit preferences for one of the two systems.

Because interleaving yields paired binary outcomes indicating which system wins for each impression, we assessed statistical significance using a two-sided binomial sign test under the null hypothesis of equal user preference. In order to compare multiple system pairs, we used the Holm–Bonferroni correction with a significance level of $\alpha = 0.01$.

5.3 Results

Table 2 reports the results from the interleaving experiments for each system pair. The table contains the number of wins for each system of the pair and the

p-value indicating the significance of the comparison. Significantly better performing systems are highlighted in bold. The results show that the baseline recommender based on term similarity received fewer clicks than other approaches, demonstrating the superiority of embedding-based recommendation methods. Although the click-path recommender (CPR) surpassed the term-similarity baseline, it remained less effective than the embedding-based similarity models.

The comparisons between embedding-based similarity models demonstrate that the nomic model produced the most favorable recommendations for GESIS Search users, while recommenders using the MiniLM and mpnet-base models performed very similarly, with no significant difference. According to the results, the overall ranking of recommender effectiveness can be summarized as follows: nomic $\succ$ MiniLM $\equiv$ mpnet-base $\succ$ CPR $\succ$ term-similarity.

We further analyzed the recommendation performance of systems across scholarly categories of GESIS Search. To this end, we identified the clicked item categories and calculated the win-lose counts for each category separately (see **Table 3**). The table clearly shows that recommendations for Publications received far more clicks than the other scholarly types. In contrast, very few clicks are directed to Instruments & Tools category. The term-similarity recommender performed particularly poorly for Publication category compared to its overall performance, as indicated by the shades of red in the table (i.e., rows 1, 2, 3, and 7). In contrast, the same approach achieved noticeably stronger results for Research Data and Variable categories, where its performance improved by approximately 15-20% relative to the overall results. In these categories, the term-similarity recommender even outperformed the click-path recommender. A similar improvement in these two categories was also observed for CPR. Among all categories, the GESIS Library results aligned most closely with the overall evaluation results from Table 2.

Comparison Pair $A \succ B$	**A wins**	**B wins**	**#queries**	**p-value**
term-sim $\succ$ **nomic**	271 (25.4%)	**797 (74.6%)**	1068	**4.5x10^{-15}**
term-sim $\succ$ **minilm**	340 (30.8%)	**765 (69.2%)**	1105	**3.9x10^{-15}**
term-sim $\succ$ **mpnet-base**	312 (29.8%)	**735 (70.2%)**	1047	**5x10^{-15}**
nomic $\succ$ minilm	**769 (56.6%)**	590 (43.4%)	1359	**1.3x10^{-6}**
nomic $\succ$ mpnet-base	**767 (55.5%)**	615 (44.5%)	1382	**4.8x10^{-5}**
mpnet-base $\succ$ minilm	606 (49.7%)	613 (50.3%)	1219	0.84
CPR $\succ$ term-sim	**602 (54.9%)**	495 (45.1%)	1097	**0.0014**
CPR $\succ$ **nomic**	339 (28.7%)	**844 (71.3%)**	1183	**5.1x10^{-15}**
CPR $\succ$ **minilm**	349 (29.3%)	**844 (70.7%)**	1193	**4.1x10^{-15}**
CPR $\succ$ **mpnet-base**	344 (30.9%)	**768 (69.1%)**	1112	**3.8x10^{-15}**

Table 2: Pairwise comparison of recommender systems. Statistically significant p-values are highlighted in bold.

	Publications		Research data		Variables		Instruments		GESIS Lib	
term-sim X nomic	85 (14.5%)	501 (85.5%)	138 (41.8%)	192 (58.2%)	33 (37.1%)	56 (62.9%)	2 (25.0%)	6 (75.0%)	13 (23.6%)	42 (76.4%)
term-sim X minilm	118 (19.5%)	488 (80.5%)	156 (48.0%)	169 (52.0%)	43 (43.9%)	55 (56.1%)	5 (45.5%)	6 (54.5%)	18 (27.7%)	47 (72.3%)
term-sim X mpnet-base	94 (17.9%)	430 (82.1%)	155 (44.2%)	196 (55.8%)	36 (39.1%)	56 (60.9%)	3 (42.9%)	4 (57.1%)	24 (32.9%)	49 (67.1%)
nomic X minilm	484 (57.8%)	354 (42.2%)	162 (53.5%)	141 (46.5%)	46 (64.8%)	25 (35.2%)	5 (62.5%)	3 (37.5%)	72 (51.8%)	67 (48.2%)
nomic X mpnet-base	492 (57.3%)	366 (42.7%)	162 (50.2%)	161 (49.8%)	37 (49.3%)	38 (50.7%)	6 (75.0%)	2 (25.0%)	70 (59.3%)	48 (40.7%)
mpnet-base X minilm	396 (49.1%)	410 (50.9%)	129 (55.1%)	105 (44.9%)	24 (48.0%)	26 (52.0%)	3 (60.0%)	2 (40.0%)	54 (43.5%)	70 (56.5%)
CPR X term-sim	319 (68.0%)	150 (32.0%)	187 (41.2%)	267 (58.8%)	42 (48.8%)	44 (51.2%)	10 (76.9%)	3 (23.1%)	44 (58.7%)	31 (41.3%)
CPR X nomic	181 (25.5%)	529 (74.5%)	89 (34.0%)	173 (66.0%)	27 (39.7%)	41 (60.3%)	6 (40.0%)	9 (60.0%)	36 (28.1%)	92 (71.9%)
CPR X minilm	187 (26.4%)	522 (73.6%)	117 (39.4%)	180 (60.6%)	18 (30.0%)	42 (70.0%)	6 (35.3%)	11 (64.7%)	21 (19.1%)	89 (80.9%)
CPR X mpnet-base	179 (27.9%)	462 (72.1%)	110 (38.5%)	176 (61.5%)	20 (33.3%)	40 (66.7%)	7 (63.6%)	4 (36.4%)	28 (24.6%)	86 (75.4%)

Table 3: Category-specific pairwise comparison of recommender systems. For each pair and category, two cells contain the number of wins and the win rate for systems in the corresponding pair. Cell colors indicate percentage change from the overall results (i.e., Table 2): decreases (red), increases (green), and near-zero changes (yellow).

6 Discussion and Limitations

In our evaluation framework, we compared five recommender systems through ten pairwise comparisons, collecting at least 1,000 clicks per comparison and 11,765 clicks in total. Given the current daily traffic of GESIS Search with 30k document views per week, we observed an average of around 500 clicks per week in the recently introduced recommendation section. Consequently, evaluating all five systems required approximately 23 weeks. We also observed that comparisons involving stronger systems accumulated clicks more quickly, whereas experiments with weaker recommenders required substantially more time to reach the same threshold. This suggests that users are less likely to engage with low-quality recommendations and may ignore them altogether. Despite the long evaluation window of 23 weeks, several actions can be taken in future iterations to optimize the evaluation framework further. First, instead of using pairwise interleaving, multi-interleaving algorithms allowing comparison of more than two systems in a single result list [71] can provide efficiency in evaluating more systems with fewer comparisons. Second, poor-performing recommenders can be filtered out through quick offline evaluation, keeping the number of systems deployed to online experiments minimal and ensuring only the top-performing models are deployed to live production traffic. Lastly, as users adopt the recently introduced recom-

mendation section, the increased click-rate will naturally shorten the evaluation phase.

Our evaluation results satisfy Strong Stochastic Transitivity (SST) [42]: for any three systems A, B, and C in our set, if the win probabilities $P(A, B) \geq 0.5$ and $P(B, C) \geq 0.5$, then $P(A, C) \geq \max[P(A, B), P(B, C)]$. This property indicates that our online evaluation setup produces consistent rankings, and the magnitude of preference increases as the performance gap between systems grows. From a practical point of view, this property confirms that a set of n systems can be reliably ranked with fewer comparisons (i.e., $O(n \log n)$), rather than exhaustively comparing all pairs (i.e., $\binom{n}{2}$) [63, 75]. This substantially reduces the total evaluation cost and user traffic required to determine system ranking.

The term-based similarity recommender is the simplest approach among the five models and serves as the baseline in our experiments. Because it relies on exact-term matching (with n-gram analysis), it cannot capture semantic similarity between documents with different wording but the same concepts. As a result, it is better suited to known-item search scenarios (e.g., dataset search [13]) than to exploratory search. In line with earlier work [12, 22], our results show that term-based methods are outperformed by semantic similarity approaches in exploratory recommendation settings. Nevertheless, this approach has clear strengths in terms of efficiency. It operates directly on existing textual metadata without requiring vector computation, storage, or approximate nearest neighbor search, all of which can be computationally demanding. Especially in large systems with frequently updated indices, operational costs rise. That makes storage of embedding vectors, and vector-space search, more labor- and infrastructure-intensive. On the other hand, the term-based similarity approach operates on textual metadata already in the index, making it lightweight and fast compared to the other techniques. Overall, while term-based recommenders are lightweight and generally effective, they underperform more advanced methods in click-through rates.

The click-path recommender (CPR) achieved the second-best performance overall, outperforming the term-based baseline but falling short of embedding-based models. Although CPR also utilizes embeddings (from the nomic model), it differs fundamentally in its objective: rather than recommending items most similar to a target item, it predicts the next item based on users' historical click paths within a session. In this sense, it models sequential navigation behavior rather than item similarity. Our results suggest that users in the academic domain tend to remain focused on a specific topic and prefer recommendations that are closely related to their current item, rather than following navigation patterns from past sessions. Additionally, CPR was trained on search logs rather than recommendation logs, which may limit its effectiveness in this context. Popularity bias in the training data may further skew its predictions toward frequently clicked items. Prior work has also shown that RNN-based session recommenders struggle in noisy scenarios, such as short sessions [51] or shifting user intent [52, 83]. Furthermore, preprocessing steps during the creation of training data likely penalized the CPR model. Because users iteratively reformulate their

queries during a session in order to improve their search, resetting the session sequences after each query reformulation might have prevented the model from capturing complex information-seeking patterns. Overall, CPR appears less well-suited to our use case.

Semantic similarity approaches achieved the best performance, consistently attracting more clicks than other methods. Among the three embedding models evaluated, performance varied noticeably, with the nomic model achieving the highest click rates. This may be caused by several factors: (i) it is based on a more recent architecture and the model was introduced lately in 2025, (ii) its Mixture-of-Experts (MoE) design has shown strong performance in multilingual settings [33,47], and (iii) it has a relatively large size with over 450M parameters. In addition, the model was fine-tuned for retrieval tasks, closely matching our recommendation scenario [58]. In contrast, the two sentence-transformer models (mpnet-base and MiniLM) produced nearly identical results, with no statistically significant difference. Given that MiniLM uses embeddings of half the dimensionality (384 vs. 768), it presents a more efficient model with comparable performance [67]. Overall, semantic similarity methods appear simple yet robust in recommending scholarly content.

Our category-specific analysis (cp. Table 3) reveals differences in user interaction across information types. In particular, Publication recommendations receive substantially more clicks than Instruments & Tools, which is likely associated with differences in category popularity within GESIS Search. As expected, users mainly engage with Publications and Research Data. Interestingly, the term-based recommender performs relatively better in Research Data and Variable categories, even surpassing CPR. This matches findings by Carevic et al. [13], who show that dataset search often resembles known-item search, in which exact-term matching may perform better. While we observe larger fluctuations in performance for the Instruments & Tools category, the limited number of clicks prevents us from drawing conclusions. Overall, although user behavior varies across categories, the relative ranking of the recommenders remains largely stable.

A main limitation of our study is its reliance on click-based evaluation. Our approach assumes that user clicks signal relevance between the target and recommended items; however, clicks may also reflect curiosity, accidental interactions, or brief inspections. To obtain a more robust assessment, click-based metrics should be complemented with additional signals such as dwell time or downstream engagement [21, 72].

Several takeaways emerge from this work. First, evaluating a diverse set of recommendation approaches is important for identifying the most suitable method for a given use case. Second, more complex models do not necessarily yield better outcomes; simpler methods may provide competitive performance at a fraction of the cost (e.g. semantic similarity vs. CPR) [54]. While the models differ in operational cost or deployment complexity, all implemented recommendation systems exhibit comparable real-time inference speeds, with satisfactory response times remaining below the noticeable threshold for end-

users. Finally, domain characteristics and information types can affect users' information-seeking behavior and should be carefully considered when designing recommender systems. These takeaways may inform the development of recommendation features in other academic search platforms.

7 Conclusion

In this study, we developed and evaluated five different recommender systems for the social science search platform GESIS Search using the STELLA framework. By testing these systems with real users in real-world settings, we identified that semantic similarity models, especially the nomic model, are the most effective at capturing user interest in an academic search scenario.

We also found that the "best" approach can depend on what is being searched. For Research Data and Variables, simpler term-based methods remain competitive because users in these categories often look for specific, known items. However, for general browsing and publications, users clearly prefer the semantic connections provided by embedding-based models.

In addition to its practical contributions to GESIS Search, this study opens several paths for future work. Planned work includes building a multi-category test collection based on the collected click data, enabling offline evaluation of academic recommender systems in a category-aware setting. This resource will be continuously updated as new interaction data becomes available. Furthermore, we aim to develop new recommendation models customized to our platform, potentially including category-specific approaches.

Overall, this research demonstrates the utility of online evaluation for developing better services and for understanding users' information-seeking behavior in academic search systems. By exploring diverse techniques and addressing the specific needs of various source types, platforms like GESIS Search can better support researchers in discovering relevant scholarly content.

Acknowledgments. This research has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) in the project STELLA II (Infrastructures for Living Labs, project number 407518790).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Alhoori, H., Furuta, R.: Recommendation of scholarly venues based on dynamic user interests. Journal of Informetrics **11**(2), 553–563 (2017)
2. Balog, K., Flekova, L., Hagen, M., Jones, R., Potthast, M., Radlinski, F., Sanderson, M., Vakulenko, S., Zamani, H.: Common conversational community prototype: Scholarly conversational assistant. arXiv preprint arXiv:2001.06910 (2020), `https://arxiv.org/pdf/2001.06910`

3. Beel, J., Collins, A., Kopp, O., Dietz, L.W., Knoth, P.: Online evaluations for everyone: Mr. dlib's living lab for scholarly recommendations. In: European Conference on Information Retrieval. pp. 213–219. Springer (2019)
4. Beel, J., Genzmehr, M., Langer, S., Nürnberger, A., Gipp, B.: A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation. In: Proceedings of the international workshop on reproducibility and replication in recommender systems evaluation. pp. 7–14 (2013)
5. Beel, J., Gipp, B., Langer, S., Breitinger, C.: Paper recommender systems: a literature survey. International Journal on Digital Libraries **17**(4), 305–338 (2016)
6. Beel, J., Langer, S.: A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems. In: International conference on theory and practice of digital libraries. pp. 153–168. Springer (2015)
7. Beel, J., Langer, S., Genzmehr, M., Nürnberger, A.: Persistence in recommender systems: giving the same recommendations to the same users multiple times. In: International Conference on Theory and Practice of Digital Libraries. pp. 386–390. Springer (2013)
8. Beel, J., Langer, S., Nürnberger, A., Genzmehr, M.: The impact of demographics (age and gender) and other user-characteristics on evaluating recommender systems. In: International conference on theory and practice of digital libraries. pp. 396–400. Springer (2013)
9. Brack, A., Hoppe, A., Ewerth, R.: Citation recommendation for research papers via knowledge graphs. In: International Conference on Theory and Practice of Digital Libraries. pp. 165–174. Springer (2021)
10. Breuer, T., Schaer, P., Tavakolpoursaleh, N., Schaible, J., Wolff, B., Müller, B.: Stella: Towards a framework for the reproducibility of online search experiments. OSIRRC@ SIGIR **2409**, 8–11 (2019)
11. Brickley, D., Burgess, M., Noy, N.: Google dataset search: Building a search engine for datasets in an open web ecosystem. In: The world wide web conference. pp. 1365–1375 (2019)
12. Capelle, M., Frasincar, F., Moerland, M., Hogenboom, F.: Semantics-based news recommendation. In: Proceedings of the 2nd international conference on web intelligence, mining and semantics. pp. 1–9 (2012)
13. Carevic, Z., Roy, D., Mayr, P.: Characteristics of dataset retrieval sessions: experiences from a real-life digital library. In: International Conference on Theory and Practice of Digital Libraries. pp. 185–193. Springer (2020)
14. Charalampous, A., Knoth, P.: Classifying document types to enhance search and recommendations in digital libraries. In: International Conference on Theory and Practice of Digital Libraries. pp. 181–192. Springer (2017)
15. Cleverdon, C.: The cranfield tests on index language devices. In: Aslib proceedings. vol. 19, pp. 173–194. MCB UP Ltd (1967)
16. Dehghani Champiri, Z., Asemi, A., Siti Salwah Binti, S.: Meta-analysis of evaluation methods and metrics used in context-aware scholarly recommender systems. Knowledge and Information Systems **61**(2), 1147–1178 (2019)
17. Ekwurzel, D.: The history and scope of the american economic association's econlit and the economic literature index. Publishing Research Quarterly **11**(3), 104–108 (1995)
18. Entrup, E., Eppelin, A., Ewerth, R., Hartwig, J., Tullney, M., Wohlgemuth, M., Hoppe, A.: B! son: A tool for open access journal recommendation. In: International Conference on Theory and Practice of Digital Libraries. pp. 357–364. Springer (2022)

19. Entrup, E., Ewerth, R., Hoppe, A.: A comparison of automated journal recommender systems. In: International Conference on Theory and Practice of Digital Libraries. pp. 230–238. Springer (2023)
20. European Organization For Nuclear Research, OpenAIRE: Zenodo (2013). https://doi.org/10.25495/7GXK-RD71, https://www.zenodo.org/
21. Fox, S., Karnawat, K., Mydland, M., Dumais, S., White, T.: Evaluating implicit measures to improve web search. ACM Transactions on Information Systems (TOIS) **23**(2), 147–168 (2005)
22. Frasincar, F., IJntema, W., Goossen, F., Hogenboom, F.: A semantic approach for news recommendation. In: Business Intelligence Applications and the Web: Models, Systems and Technologies, pp. 102–121. IGI Global Scientific Publishing (2012)
23. Gasparyan, A.Y., Yessirkepov, M., Voronov, A.A., Trukhachev, V.I., Kostyukova, E.I., Gerasimov, A.N., Kitas, G.D.: Specialist bibliographic databases. Journal of Korean Medical Science **31**(5), 660–673 (2016)
24. Ginsparg, P.: Lessons from arxiv's 30 years of information sharing. Nature Reviews Physics **3**(9), 602–603 (2021)
25. Gupta, V., Pandey, S.R.: Recommender systems for digital libraries: a review of concepts and concerns. Library Philosophy and Practice pp. 1–9 (2019)
26. Hahnel, M.: Exclusive: figshare a new open data project that wants to change the future of scholarly publishing. Impact of Social Sciences Blog (2012), https://researchonline.lse.ac.uk/id/eprint/51893/1/blogs.lse.ac.uk-Exclusive_figshare_a_new_open_data_project_that_wants_to_change_the_future_of_scholarly_publishing.pdf
27. Hanyurwimfura, D., Bo, L., Havyarimana, V., Njagi, D., Kagorora, F.: An effective academic research papers recommendation for non-profiled users. International Journal of Hybrid Information Technology **8**(3), 255–272 (2015)
28. Harman, D.: Overview of the first trec conference. In: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 36–47 (1993)
29. Hassan, H.A.M., Sansonetti, G., Gasparetti, F., Micarelli, A., Beel, J.: Bert, elmo, use and infersent sentence encoders: The panacea for research-paper recommendation? In: RecSys (Late-Breaking Results). pp. 6–10 (2019)
30. Hienert, D., Kern, D., Boland, K., Zapilko, B., Mutschke, P.: A digital library for research data and related information in the social sciences. In: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL). pp. 148–157. IEEE (2019)
31. Hristakeva, M., Kershaw, D., Rossetti, M., Knoth, P., Pettit, B., Vargas, S., Jack, K.: Building recommender systems for scholarly information. In: Proceedings of the 1st workshop on scholarly web mining. pp. 25–32 (2017)
32. Hurtado Martín, G., Schockaert, S., Cornelis, C., Naessens, H.: Metadata impact on research paper similarity. In: International Conference on Theory and Practice of Digital Libraries. pp. 457–460. Springer (2010)
33. Ijebu, F.F., Liu, Y., Sun, C., Jere, N., Mienye, I.D., Usip, P.U.: Ensemble answer selection leveraging cross-lingual dealignment for improved question answering with mixture-of-experts setup. IEEE Open Journal of the Computer Society (2025)
34. Jacsó, P.: Google scholar: the pros and the cons. Online information review **29**(2), 208–214 (2005)
35. Jensen, E.C., Beitzel, S.M., Chowdhury, A., Frieder, O.: Repeatable evaluation of search services in dynamic environments. ACM Transactions on Information Systems (TOIS) **26**(1), 1–es (2007)

36. Joachims, T.: Optimizing search engines using clickthrough data. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 133–142 (2002)
37. Kaur, H., Downey, D., Singh, A., Cheng, E.Y.Y., Weld, D., Bragg, J.: Feedlens: polymorphic lenses for personalizing exploratory search over knowledge graphs. In: Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology. pp. 1–15 (2022)
38. Keller, J., Breuer, T., Schaer, P.: Evaluation of temporal change in ir test collections. In: Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval. pp. 3–13 (2024)
39. Keshavarz, S., Honarvar, A.R.: A parallel paper recommender system in big data scholarly. In: International conference on electrical engineering and computer (2015)
40. Kohavi, R., Longbotham, R., Sommerfield, D., Henne, R.M.: Controlled experiments on the web: survey and practical guide. Data mining and knowledge discovery **18**(1), 140–181 (2009)
41. Kong, X., Mao, M., Wang, W., Liu, J., Xu, B.: Voprec: Vector representation learning of papers with text information and structural identity for recommendation. IEEE Transactions on Emerging Topics in Computing **9**(1), 226–237 (2018)
42. Kozielecki, J.: Psychological decision theory, vol. 24. Springer Science & Business Media (1982)
43. Krämer, T., Papenmeier, A., Carevic, Z., Kern, D., Mathiak, B.: Data-seeking behaviour in the social sciences. International Journal on Digital Libraries **22**(2), 175–195 (2021)
44. Krestel, R., Smyth, P.: Recommending patents based on latent topics. In: Proceedings of the 7th ACM Conference on Recommender Systems. pp. 395–398 (2013)
45. Küçüktunç, O., Saule, E., Kaya, K., Çatalyürek, Ü.V.: Towards a personalized, scalable, and exploratory academic recommendation service. In: Proceedings of the 2013 IEEE/ACM international conference on Advances in social networks analysis and mining. pp. 636–641 (2013)
46. Ley, M.: The dblp computer science bibliography: Evolution, research issues, perspectives. In: International symposium on string processing and information retrieval. pp. 1–10. Springer (2002)
47. Li, C., Deng, Y., Zhang, J., Zong, C.: Group then scale: Dynamic mixture-of-experts multilingual language model. In: Findings of the Association for Computational Linguistics: ACL 2025. pp. 1730–1754 (2025)
48. Li, X., Chen, Y., Pettit, B., Rijke, M.D.: Personalised reranking of paper recommendations using paper content and user behavior. ACM Transactions on Information Systems (TOIS) **37**(3), 1–23 (2019)
49. Li, Y., Wang, R., Nan, G., Li, D., Li, M.: A personalized paper recommendation method considering diverse user preferences. Decision Support Systems **146**, 113546 (2021)
50. Li, Z., Rainer, A.: Academic search engines: constraints, bugs, and recommendations. In: Proceedings of the 13th International Workshop on Automating Test Case Design, Selection and Evaluation. pp. 25–32 (2022)
51. Li, Z., Yang, C., Chen, Y., Wang, X., Chen, H., Xu, G., Yao, L., Sheng, M.: Graph and sequential neural networks in session-based recommendation: A survey. ACM Computing Surveys **57**(2), 1–37 (2024)
52. Liu, Q., Zeng, Y., Mokhosi, R., Zhang, H.: Stamp: short-term attention/memory priority model for session-based recommendation. In: Proceedings of the 24th ACM

SIGKDD international conference on knowledge discovery & data mining. pp. 1831–1839 (2018)
53. Lossau, N., Rahmsdorf, S., Pieper, D., Summann, F.: Bielefeld academic search engine (base) an end-user oriented institutional repository search service. Library Hi Tech **24**(4), 614–619 (2006)
54. Ludewig, M., Jannach, D.: Evaluation of session-based recommendation algorithms: M. ludewig, d. jannach. User Modeling and User-Adapted Interaction **28**(4), 331–390 (2018)
55. Ma, Z., Liu, B., Li, Y., Zhang, X.: Crs-mentor: A conversational recommender system for large language model-enhanced academic resource mentorship. In: 2025 8th International Symposium on Big Data and Applied Statistics (ISBDAS). pp. 42–49. IEEE (2025)
56. Medrek, J., Otto, C., Ewerth, R.: Recommending scientific videos based on metadata enrichment using linked open data. In: International Conference on Theory and Practice of Digital Libraries. pp. 286–292. Springer (2018)
57. Nandy, P., Venugopalan, D., Lo, C., Chatterjee, S.: A/b testing for recommender systems in a two-sided marketplace. Advances in Neural Information Processing Systems **34**, 6466–6477 (2021)
58. Nussbaum, Z., Duderstadt, B.: Training sparse mixture of experts text embedding models (2025), https://arxiv.org/abs/2502.07972
59. Ong, D., Truong, Q.T., Lauw, H.W.: Cornac-ab: An open-source recommendation framework with native a/b testing integration. In: Companion Proceedings of the ACM Web Conference 2024. pp. 1027–1030 (2024)
60. Pera, M.S., Ng, Y.K.: A personalized recommendation system on scholarly publications. In: Proceedings of the 20th ACM international conference on Information and knowledge management. pp. 2133–2136 (2011)
61. Peters, C.: Cross-Language Information Retrieval and Evaluation: Workshop of Cross-Language Evaluation Forum, CLEF 2000, Lisbon, Portugal, September 21-22, 2000, Revised Papers, vol. 2069. Springer (2003)
62. Radlinski, F., Craswell, N.: Optimized interleaving for online retrieval evaluation. In: Proceedings of the sixth ACM international conference on Web search and data mining. pp. 245–254 (2013)
63. Radlinski, F., Kurup, M., Joachims, T.: How does clickthrough data reflect retrieval quality? In: Proceedings of the 17th ACM conference on Information and knowledge management. pp. 43–52 (2008)
64. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. 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). pp. 3982–3992 (2019)
65. Risch, J., Krestel, R.: What should i cite? cross-collection reference recommendation of patents and papers. In: International Conference on Theory and Practice of Digital Libraries. pp. 40–46. Springer (2017)
66. Sanderson, M.: Test collection based evaluation of information retrieval systems. Foundations and Trends® in Information Retrieval **4**(4), 247–375 (2010)
67. SBERT Pretrained Models. https://www.sbert.net/docs/sentence_transformer/pretrained_models.html, accessed: 2026-04-30
68. Schaer, P., Schaible, J., Castro, L.J.: Living lab evaluation for life and social sciences search platforms-lilas at clef 2021. In: European Conference on Information Retrieval. pp. 657–664. Springer (2021)
69. Schaer, P., Schaible, J., Müller, B.: Living labs for academic search at clef 2020. In: European Conference on Information Retrieval. pp. 580–586. Springer (2020)

70. Schonfeld, R.C.: Jstor: a history (2012)
71. Schuth, A., Sietsma, F., Whiteson, S., Lefortier, D., de Rijke, M.: Multileaved comparisons for fast online evaluation. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. pp. 71–80 (2014)
72. Sharma, H., Jansen, B.J.: Automated evaluation of search engine performance via implicit user feedback. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 649–650 (2005)
73. Smeaton, A.F., Callan, J.: Personalisation and recommender systems in digital libraries. International Journal on digital libraries **5**(4), 299–308 (2005)
74. Sterling, J.A., Montemore, M.M.: Combining citation network information and text similarity for research article recommender systems. IEEE Access **10**, 16–23 (2021)
75. Szörényi, B., Busa-Fekete, R., Paul, A., Hüllermeier, E.: Online rank elicitation for plackett-luce: A dueling bandits approach. Advances in neural information processing systems **28** (2015)
76. Valtysson, B.: Europeana: The digital construction of europe's collective memory. Information, Communication & Society **15**(2), 151–170 (2012)
77. Voorhees, E.M., Harman, D.K., et al.: TREC: Experiment and evaluation in information retrieval, vol. 63. MIT press Cambridge (2005)
78. Wang, Q., Li, W., Zhang, X., Lu, S.: Academic paper recommendation based on community detection in citation-collaboration networks. In: Asia-Pacific web conference. pp. 124–136. Springer (2016)
79. Wang, X., Duong-Trung, N., Bhoyar, R.R., Jose, A.M.: Llm-based literature recommender system in higher education: A case study of supervising students' term papers. Computers & Education: Artificial Intelligence **10** (2025)
80. White, J.: Pubmed 2.0. Medical reference services quarterly **39**(4), 382–387 (2020)
81. White, J.: Acm digital library enhancements. Communications of the ACM **42**(4), 30–31 (1999)
82. Wu, L., Li, Z., Zhao, H., Huang, Z., Han, Y., Jiang, J., Chen, E.: Supporting your idea reasonably: A knowledge-aware topic reasoning strategy for citation recommendation. IEEE Transactions on Knowledge and Data Engineering **36**(8), 4275–4289 (2024)
83. Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., Tan, T.: Session-based recommendation with graph neural networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 346–353 (2019)
84. Yang, Z., Davison, B.D.: Venue recommendation: Submitting your paper with style. In: 2012 11th international conference on machine learning and applications. vol. 1, pp. 681–686. IEEE (2012)
85. Zangerle, E., Bauer, C.: Evaluating recommender systems: survey and framework. ACM computing surveys **55**(8), 1–38 (2022)
86. Zimmermann, C.: Academic rankings with repec. Econometrics **1**(3), 249–280 (2013)