Journal of Intelligent Information Systems (2026) 64:349–377
https://doi.org/10.1007/s10844-025-01004-9

RESEARCH

Leveraging knowledge graphs and LLMs for content-based reviewer assignment

Farid Bagheri[1] · **Davide Buscaldi**[2] · **Diego Reforgiato Recupero**[1]

Received: 24 June 2025 / Revised: 27 October 2025 / Accepted: 28 October 2025 /
Published online: 10 November 2025
© The Author(s) 2025

Abstract
The growing volume of academic submissions in recent years highlighted the need for scalable and accurate reviewer assignment systems, able to go beyond techniques based on manual processes and basic keyword matching. We propose a novel pipeline that integrates Knowledge Graphs (KGs) and Large Language Models (LLMs) to automate and enhance the reviewer assignment process. Our method extracts meaningful representations of papers and reviewer expertise using Open Information Extraction, the Computer Science Ontology classifier, and GLiNER to build KGs from research content. LLMs are employed to generate targeted keywords through prompt-based synthesis, refining both paper and reviewer profiles. The assignment relies on a hybrid similarity metric combining Cosine and Jaccard similarities to capture both lexical and semantic alignment. We evaluate the pipeline using standard metrics such as Mean Reciprocal Rank, Mean Average Precision, and Precision at K, on a dataset in the Computer Science domain, demonstrating its effectiveness in aligning submissions with appropriate reviewers. This approach offers a scalable and adaptive solution to the complexities of modern peer review.

Keywords Reviewer assignment · Semantic web · Recommender systems · Large language models · Knowledge graphs

Farid Bagheri, Davide Buscaldi and Diego Reforgiato Recupero contributed equally to this work.

✉ Diego Reforgiato Recupero
diego.reforgiato@unica.it

Farid Bagheri
farid.bagheri@unica.it

Davide Buscaldi
davide.buscaldi@lipn.univ-paris13.fr

[1] Department of Mathematics and Computer Science, University of Cagliari, Via Ospedale 72, Cagliari 09042, Italy

[2] Laboratoire d'Informatique de Paris Nord, Sorbonne Paris Nord University, 99 Av. Jean Baptiste Clement, Villetaneuse 93430, Paris, France

1 Introduction

The exponential growth in academic submissions has intensified the need for efficient, accurate, and scalable reviewer assignment systems (Li et al., 2025). Conferences and journals in fields such as computer science, medicine, and engineering now receive thousands of submissions per cycle, making manual reviewer assignment increasingly impractical (Zhao and Zhang, 2022). Traditional approaches, often relying on manually curated reviewer pools, keyword matching, or bidding systems, struggle to keep up with this volume (Bhaisare and Bharati, 2024). Moreover, these methods are prone to various limitations, including subjectivity, inconsistent coverage of emerging research areas, and an inability to capture the complex semantics of modern scientific writing (Aksoy et al., 2023).

Several automated approaches have been proposed to alleviate this burden (Aksoy et al., 2023). For instance, the Toronto Paper Matching System (TPMS) (Charlin and Zemel, 2013) uses latent semantic analysis to represent papers and reviewer profiles in a shared topic space, enabling similarity-based matching. Other methods have incorporated topic modeling techniques like Latent Dirichlet Allocation (LDA) to identify thematic structures in submissions and reviewer publications (Madzík and Falát, 2022). While these approaches improve scalability, they often fall short in handling the dynamic and interdisciplinary nature of current research. Their reliance on bag-of-words representations or shallow semantic features can lead to inaccurate or overly generic matches (Qader et al., 2019).

More recently, deep learning models and contextual embeddings have been introduced to better capture the meaning of research texts (Sarzynska-Wawer et al., 2021). Systems such as OpenReview (Zhang et al., 2025) have experimented with BERT-based embeddings for paper and reviewer representations, improving the granularity of the match. However, these models often operate as black boxes, offering limited interpretability and lacking structured insights into the content being matched (Sun et al., 2024). Furthermore, they rarely incorporate domain-specific knowledge or the explicit relationships between concepts that could enhance the understanding of both papers and reviewer expertise.

In this work, we propose a novel pipeline for reviewer assignment that leverages KGs and LLMs to enhance the matching of candidate papers with suitable reviewers. Our approach for KGs builds upon three key components for extracting meaningful representations of papers and reviewers: Open Information Extraction (OpenIE), graph-based linguistic Named Entity Recognition (GLiNER) (Zaratiana et al., 2024), and the Computer Science Ontology (CSO) Classifier (Salatino et al., 2019, 2022) (see Sections 4.2.3 and 4.2.4). OpenIE is employed to extract triples (subject, relation, object) from the title, abstract, and summary of abstracts, while on the other hand, GLINER and the CSO classifier are used to extract research topics and entities related to the papers from the output of OpenIE. These extracted elements are then used to construct KGs, which serve as structured representations of expertise and paper content. In parallel to the KG-based representation, we employ LLMs to extract keyword-based semantic profiles from papers and reviewer publications, which are then used to compute similarity based on textual content. For candidate papers, the LLM synthesizes keywords to identify potential reviewers, while for reviewer profiles, it distills author expertise from their publication history. The core matching process involves computing similarity scores between candidate papers and reviewer profiles using Cosine similarity, Jaccard similarity, and an averaged metric that combines both. This dual-measure approach ensures that both lexical overlap and semantic relevance are considered, thus improving

the accuracy of the assignment. To validate the efficacy of our method, we evaluate the reviewer assignment results against actual reviewer allocations using established metrics such as Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), and Precision at K (P@K). By integrating advanced Natural Language Processing (NLP) techniques, KGs construction, and LLM-driven keyword extraction, our pipeline represents a significant step forward in automating reviewer assignment. This work not only addresses the limitations of conventional methods but also provides a scalable framework that can adapt to the evolving demands of academic peer review.

To address these challenges, we design a pipeline that leverages two complementary perspectives: (i) KGs constructed from triples extracted with OpenIE and enriched with the CSO Classifier and GLiNER topics, and (ii) keywords generated by LLMs from paper titles, abstracts, and summaries. These two tracks are applied independently to model reviewer expertise and candidate submissions, allowing us to compare their relative effectiveness in the reviewer assignment task. Prior work has explored joint KG–LLM frameworks (Pan et al., 2024; Mariotti et al., 2024; Giarelis et al., 2024; Marchesin et al., 2025), showing benefits for tasks such as natural language understanding, fact-checking, and data quality. In contrast, we treat KGs and LLMs as separate tracks to enable a clean comparative evaluation.

The remainder of this paper is organized as follows. Section 2 provides a concise overview of prior research and related works in the domain of reviewer assignment systems, contextualizing the current study within the existing literature. Section 3 presents the targeted research task and elucidates the objectives of this work, including the key challenges and contributions of the proposed methodology. Section 4 details the proposed reviewer assignment pipeline. This section is subdivided into four subsections: Section 4.1 describes the datasets utilized in this study, including their sources, characteristics, and relevance to the task. Section 4.2 outlines the preprocessing steps applied to the raw data to before building KGs and also explains the LLM-based approach. Section 4.3 discusses the feature engineering strategies employed to extract meaningful representations of reviewers and submissions for constructing the KGs. Section 4.4 explains the similarity computation methods and the methodology employed for matching potential reviewers to submissions. Then, Section 5 elaborates on the evaluation framework, including the experimental setup, baseline methods, and performance metrics used to assess the proposed system. Experimental results are reported and analyzed in Section 6, which includes comparative analyses, statistical validations, and discussions of key findings. Finally, Section 7 concludes the paper by summarizing the study's outcomes, highlighting its theoretical and practical implications, and suggesting directions for future research.

2 Related works

The reviewer assignment problem entails allocating submitted manuscripts to suitable reviewers by matching reviewer expertise to manuscript topics, ensuring fairness, and avoiding conflicts of interest. Traditional approaches to the problem, such as manual assignments or simple keyword matching, have shown limitations in handling the growing complexity and volume of submissions. In response, recent research has introduced a range of methodologies, encompassing semantic text analysis, KGs, machine learning, and optimi-

zation techniques. This section reviews the literature, categorizing prior work based on its methodological approaches.

Initial strategies for reviewer assignment relied heavily on manual selection or bidding systems. Such methods led to workload imbalances and did not always ensure expertise alignment, as shown by authors in Mittal et al. (2019). Introducing keyword-based automation improved scalability but often failed to capture semantic depth. For instance, authors in Adebiyi et al. (2019) compared techniques including LDA and Term Frequency–Inverse Document Frequency (TF-IDF), observing that while LDA enhanced semantic alignment for interdisciplinary topics, simpler methods like TF-IDF were computationally efficient but less nuanced. Similarly, authors in Cagliero et al. (2018) noted that keyword matching alone struggled to handle conflicts of interest, potentially leading to biased or suboptimal reviewer selections.

To address these shortcomings, subsequent research turned to topic modeling, aiming to better capture latent semantics and relationships between reviewers and manuscripts. Techniques such as LDA, Latent Semantic Indexing (LSI), and TF-IDF have been explored for aligning semantic content and reviewer expertise (Adebiyi et al., 2019). Authors in Kusumawardani and Khairunnisa (2018) applied an author-topic model to assignments involving Bahasa Indonesia manuscripts, leveraging language-specific preprocessing. Others have incorporated temporal dynamics, noting that reviewer expertise evolves over time. The study in Peng et al. (2017) integrated time-aware topic modeling to emphasize recent publications, better reflecting current interests and improving assignment relevance.

KGs have emerged as valuable tools for mapping complex relationships, including those connecting authors, reviewers, institutions, and research areas. Authors in Rordorf et al. (2023) combined machine learning and KGs to manage conflicts of interest and enhance fairness, employing semantic text similarity and graph-based representations. The work performed in Yong et al. (2021), a framework utilizing KGs and rule-based matching, successfully aligned hierarchical representations of reviewer profiles with manuscript content. Moreover, experts in Xiao et al. (2022, 2023) developed a hierarchical interdisciplinary research proposal classification network, integrating semantic extraction and KGs to handle interdisciplinary proposals by predicting topic paths enriched with interdisciplinary context.

Building on these representations, neural methods such as deep learning and LLMs have been applied to further refine expertise modeling and similarity computation. Researchers in Duan et al. (2019) modeled semantic relationships between manuscripts and reviewer publications using neural networks, such as BERT, CNN, and biLSTM, to improve recommendation precision. Another work performed in Zhang et al. (2024) introduced a semantic and correlation fusion approach, integrating semantic embeddings and graph-based correlation modeling to refine reviewer-paper matching. LLMs have also been utilized to address data imbalance and improve classifier performance in domain-specific scenarios. For instance, authors in Cai et al. (2023) employed LLMs as a data augmentation tool, enhancing classification accuracy for underrepresented disciplines.

Addressing fairness and managing conflicts of interest are other considerations in the reviewer assignment problem. Authors in Rordorf et al. (2023) employed KGs to detect conflicts of interest by modeling co-authorships and institutional relationships. Fair allocation strategies have also emerged, including the Reviewer Round Robin method in Payan (2022), designed to achieve near envy-freeness by balancing reviewer workloads. Similarly, the PeerReview4All algorithm, which is carried out in Stelmakh et al. (2019), applies incre-

mental max-flow techniques to ensure max-min fairness. Other approaches include leveraging fuzzy graph connectivity, as applied in Mittal et al. (2020), or utilizing collaboration distances in academic social networks, as explored in Li et al. (2017), to enhance equity in reviewer assignments. In parallel, researchers in Nugroho et al. (2023); Cagliero et al. (2018) utilized a graph-based model to detect conflicts of interest through relationship patterns, enhancing fairness and integrity.

Comprehensive surveys, such as the dissertation in Misale and Vanwari (2017) and the analysis in Mittal et al. (2019), have reviewed techniques, limitations, and key challenges in the reviewer assignment problem. They highlight the importance of automated keyword extraction, refined expertise representation, and advanced semantic and fairness-oriented methodologies.

Recent studies have explored closer integrations of KGs and LLMs. For example, Pan et al. (2024) present a roadmap for unifying the two paradigms, while Mariotti et al. (2024) discuss how LLMs can be combined with enterprise KGs to enhance natural language understanding. Giarelis et al. (2024) propose a unified framework to support fact-checking in public deliberation, and Marchesin et al. (2025) focus on improving KG data quality using LLMs. In contrast, our approach does not fuse KGs and LLMs into a single model. Instead, we analyze them as *separate but comparable* tracks for expertise representation and reviewer assignment, highlighting their individual strengths and trade-offs.

While substantial progress has been achieved, ranging from integrating semantic text analysis and KGs to applying deep learning, significant challenges remain. These include effectively managing conflicts of interest, accommodating interdisciplinary submissions, addressing data imbalances, and accurately capturing evolving expertise. The integration of emerging techniques, such as LLM-based keyword extraction and advanced semantic understanding, presents a promising avenue for improvement. Building on these advancements, our proposed approach leverages KGs and LLM-based processing to enhance reviewer assignments' accuracy, fairness, and adaptability.

Beyond the methods reviewed above, three families frequently serve as practical baselines for reviewer assignment. First, TPMS-style systems perform lexical matching between manuscripts and reviewer profiles using TF-IDF and Cosine similarity, sometimes with heuristic normalization and stop-term handling (Charlin and Zemel, 2013). Second, embedding-based retrieval represents papers and reviewer texts with transformer encoders (e.g., SBERT) and ranks by Cosine similarity (Reimers and Gurevych, 2019; OpenReview, 2023; OpenReview: Openreview matcher, 2019; OpenReview, 2025). Third, fairness-aware assignment algorithms such as PeerReview4All optimize coverage and balance under conflict and load constraints (Stelmakh et al., 2019). These baselines inform our experimental setup and serve as comparators to the KG and LLM tracks evaluated in this work.

A recent contribution by Tong et al. (2024) introduced the Conditional Generative Adversarial Meta-Learning (CGAML) framework, addressing the limitations of existing knowledge graph completion (KGC) methods when dealing with sparse and complex data. CGAML integrates Conditional Generative Adversarial Networks (CGANs) and meta-learning to enable few-shot KGC by leveraging hierarchical background knowledge and conditional vectors that guide semantic generation. The meta-learning component adapts quickly to new relational patterns through local and global optimization strategies, while the adversarial component ensures that generated triples preserve semantic constraints and diversity. Experimental results on benchmark datasets (e.g., NELL-One, Wiki-One, and

FB15k-237) demonstrated that CGAML outperforms state-of-the-art methods in few-shot scenarios, highlighting its ability to generalize across low-frequency entities and relations. Although developed for KGC, this work provides methodological insights for reviewer-assignment problems, where meta-learning and conditional generation could support adaptive matching in domains with sparse reviewer expertise or rapidly evolving topics.

In a subsequent study, authors in Wang et al. (2025) proposed a Textual and Structural Dual Enhancement (TSDE) framework that leverages LLMs for augmenting KGs. TSDE addresses two key challenges in KGC, low-quality textual descriptions and structural sparsity, by combining LLM-based text generation with structural similarity mining. The framework employs a bidirectional depth-first sampling algorithm to extract path-based contextual information, which is then used in prompt templates guiding LLMs to generate enriched entity descriptions aligned with graph semantics. Simultaneously, cosine-based similarity computation among entities introduces synthetic "same-as" relations, densifying the graph and mitigating long-tail issues. Applied to models such as TransE, RotatE, and SimKGC, TSDE yielded significant performance gains across multiple benchmarks (FB15k-237 and WN18RR). This approach exemplifies how LLMs can serve as knowledge-enhancement tools rather than reasoning engines—a perspective directly relevant to reviewer-assignment contexts where textual and relational signals must be jointly optimized.

3 The targeted task

The targeted task of the present pipeline is to facilitate a more effective and accurate assignment of reviewers to academic manuscripts. We seek to identify expert reviewers whose research interests, methodological backgrounds, and thematic specializations align closely with the core content and intellectual scope of a given submitted paper. Formally, the task involves mapping each candidate paper to an optimized set of reviewers based on the semantic and structural features extracted from both the paper's textual content and the reviewers' publication records. This entails several integrated subtasks:

- **Text-to-knowledge conversion**: Transforming raw textual data from research abstracts and manuscripts into enriched semantic representations. Through a comprehensive preprocessing pipeline encompassing semantic triple extraction, topic enrichment, and LLM-based keyword extraction, we derive coherent knowledge structures that encapsulate the central themes and salient entities of the documents.
- **Graph-based semantic modeling**: Constructing KGs from the resulting enriched semantic triples, thus enabling a structured representation of the thematic and relational aspects of both papers and reviewers' works. Node embedding algorithms then produce dense vector representations of these graphs, enabling comparability across multiple documents.
- **Similarity computation and reviewer matching**: Employing multiple similarity measures to quantify the alignment between the thematic profiles of candidate papers and potential reviewers' research trajectories.

The end goal is an automated, data-driven mechanism that assigns each submitted manuscript to reviewers whose expertise closely resonates with the paper's content. By integrating

graph-based semantic modeling with advanced language models, this methodology ensures that the reviewer recommendation process is both contextually informed and aligned with domain-specific knowledge structures.

4 Reviewer assignment pipeline

In this study, we propose a KG-based and LLM-based approach to enhance the matching process between academic papers and potential reviewers. As indicated in Figure 1, our pipeline includes four different blocks: Data Collection, Preprocessing, Feature Engineering, and Reviewer Matching. We run two independent, non-fused pipelines that share data collection but diverge in preprocessing and feature construction.

- **(A) KG-based path:** (1) *Semantic triple extraction:* apply OpenIE to the paper text to obtain triples (s, r, o). (2) *Topic mapping and entity enrichment:* map textual spans from the OpenIE output to CSO topics to form a topic set T and also extract named entities with GLiNER to form an entity set E; (3) *KG construction:* build a graph where nodes come from subjects/objects and edges are OpenIE relations (either typed or collapsed to *related_to*). (4) *Feature construction:* (i) a discrete set of topics/entities for **Jaccard**; (ii) a vector embedding of the graph via **Node2Vec** (graph readout) for **cosine**. (5) *Similarity & ranking:* compute Jaccard, cosine, and their unweighted mean ("cosine–Jaccard average") *within this path*, then rank reviewers by the chosen score.
- **(B) LLM-keyword path.** (1) *Keyword extraction:* use LLaMA-3.2 to extract keywords from the paper's *title*, *abstract summary*, and *abstract*. (2) *Normalization:* lowercase, lemmatize, and deduplicate the keyword set. (3) *Feature construction:* (i) use keyword set directly as a set for **Jaccard**; (ii) build a keyword vector for **cosine**. (4) *Similarity & ranking:* compute cosine, Jaccard, and their unweighted mean *within this path*, and rank reviewers accordingly.

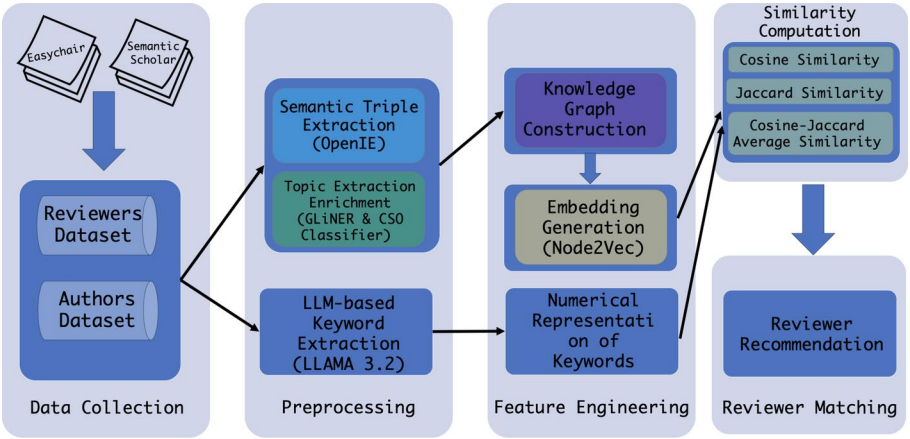

Fig. 1 Reviewer assignment pipeline

All similarities are computed within a path; there is no cross-pipeline fusion, weighting, or conflict resolution.

4.1 Data collection

Our study relies on two curated datasets, one containing submitted papers from authors and the other describing potential reviewers. These datasets are fundamental to our reviewer assignment system, providing comprehensive and structured information about submitted papers and potential reviewers. The authors' dataset, sourced from EasyChair[1], contains abstracts and metadata of research articles submitted to various conferences. In contrast, the reviewers' dataset, extracted from Semantic Scholar[2], includes detailed bibliometric information about potential reviewers, such as their publications and citation metrics. Both datasets are stored in JSON format, employing unique identifiers to ensure seamless integration and traceability across our system. Their integration forms the backbone of our reviewer assignment process, enabling a robust and scalable framework for reviewer allocation. The datasets used in this study were collected in our previous work (Bagheri et al., 2024), using publicly accessible sources[3]. The authors' dataset was extracted from EasyChair using automated tools, and the reviewers' dataset was compiled via the Semantic Scholar API. During the original data collection process, all necessary ethical guidelines and privacy considerations were followed, ensuring that no sensitive or confidential information was collected. In the current study, we use these datasets exclusively for academic research purposes. All data is anonymized, non-sensitive, and publicly available, and our use complies with the terms of service of the respective platforms. The present dataset comprises 663 papers and 524 reviewers in computer science.

4.1.1 Authors' dataset

We collected 663 papers and their assigned reviewers from 85 computer science conferences. These papers encompass a diverse range of topics within NLP and Artificial Intelligence (AI), including text summarization, sentiment analysis, KGs development, machine learning applications, and cross-lingual domain adaptation. Many entries emphasize practical applications of NLP technologies, reflecting a trend toward solving real-world problems through theoretical advancements. Each entry in the authors' dataset is uniquely identified by a dictionary key, serving as a primary identifier throughout the project files. The dataset includes comprehensive metadata for each paper: conference name, paper title, paper link, list of authors, paper abstract, and any assigned reviewers. This detailed information forms the foundational layer for evaluating and matching paper topics to reviewers based on thematic relevance.

[1] https://www.easychair.org/

[2] https://www.semanticscholar.org/

[3] https://github.com/faridbagheri/Reviewers-Recommendations-System.git

4.1.2 Reviewers' dataset

Complementing the authors' dataset is the reviewers' dataset, which centralizes reviewer-centric data linked to the reviewers assigned by EasyChair for the 663 papers previously collected. Data retrieval was performed using the Semantic Scholar Application Programming Interface (API)[4]. Each paper in the authors' dataset was typically assigned to three reviewers. For each reviewer associated with the 663 papers, we retrieved their top 20 most cited articles from Semantic Scholar, arranging them in descending order of citation count. In total, we successfully extracted information for 524 unique reviewers, which is fewer than the expected 1,989 reviewers (3 reviewers per paper $\times$ 663 papers). This discrepancy may be due to overlapping reviewer assignments, where the same reviewer was assigned to multiple papers, or limitations in data retrieval from Semantic Scholar. The reviewers' dataset is organized with each reviewer indexed by a unique tuple consisting of their author ID and name. For each reviewer, the dataset includes a list of their publications, each accompanied by details such as the number of influential citations, total citations, paper titles, abstracts, and publication links. This rich bibliometric and thematic information enables a robust assessment of reviewers' domain-specific expertise and scholarly influence. By integrating this dataset with the authors' dataset, our system performs semantic content analysis to align submission content with reviewer expertise. This data-driven approach not only streamlines the review process but also enhances the quality of peer reviews, thereby fostering academic integrity and rigor.

4.2 Preprocessing

Before constructing the KGs and performing the reviewer assignment, we have a preprocessing phase that converts raw textual content into structured, semantically rich representations. We implement five key preprocessing steps: first, semantic triple extraction by using OpenIE to isolate fundamental subject-relation-object relationships; second, topic extraction and enrichment to identify and elaborate core thematic elements; third, GLiNER to integrate linguistic and graph-based features for entity identification; fourth, the CSO classifier to extract relevant topics from the output generated by OpenIE. Finally, LLM-based keyword extraction by using LLaMA-3.2 to extract keywords related to papers. Collectively, these steps yield a coherent knowledge substrate, primed for accurate reviewer assignment.

4.2.1 Semantic triple extraction (OpenIE)

To capture essential information from the abstracts, we employed the OpenIE model[5] to extract semantic triples in the form of (subject, relation, object). OpenIE is an unsupervised method that identifies relational tuples from natural language text, enabling the extraction of structured information from unstructured data. For each abstract in both datasets, the OpenIE model processed the text and outputted a set of triples representing the key propositions expressed. This step transformed the textual content into a structured format suitable for further analysis and served as the initial seed for our knowledge-graph construction.

[4] https://api.semanticscholar.org/api-docs/

[5] https://nlp.stanford.edu/software/openie.html

OpenIE leverages syntactic parsing and semantic role labeling to identify and extract relationships, entities, and facts from text, forming the foundational components of knowledge representation.

OpenIE produces a candidate structured view, but the extracted triples often suffer from boundary errors, coarse or inconsistent relation labels, and unresolved coreference, especially in scientific text. To limit error propagation into the KG, we (i) prune malformed spans and relations with weak evidence, (ii) optionally collapse sparse or heterogeneous relations into a generic `related_to` edge, and (iii) normalize node labels through CSO/GLiNER enrichment. For downstream tasks, similarity measures (Cosine, Jaccard, or their mean) help further reduce sensitivity to noisy or spurious edges (Angeli et al., 2015).

4.2.2 Topic extraction and enrichment

Unlike traditional bag-of-words representations, which may overweight generic terms, our approach constructs a *bag-of-concepts* derived from the domain-specific KG. Concepts are disambiguated, synonym-merged, and semantically typed, ensuring that overly broad terms receive low weight and that matches remain both precise and contextually meaningful. After obtaining the semantic triples, we further enriched them by identifying relevant topics using two complementary methods: GLiNER[6] and the CSO Classifier[7], which together enhance both the semantic coverage and domain alignment of the resulting representations.

4.2.3 Graph-based Linguistic Named Entity Recognition (GLiNER)

GLiNER is an advanced tool designed for domain-specific named entity recognition (NER) (Zaratiana et al., 2024). It combines traditional entity recognition techniques with knowledge base linking to improve the identification and classification of entities within a specific domain. GLiNER utilizes domain-specific gazetteers and links recognized entities to entries in knowledge bases such as the CSO (Zaratiana et al., 2024). Unlike generic NER systems, GLiNER is optimized for low-resource or specialized domains by leveraging both linguistic cues and structured background knowledge. It employs transformer-based contextual encoders to capture semantics and dynamically integrates gazetteer features during training, which enables the detection of fine-grained scientific concepts such as tasks, methods, tools, and datasets. This hybrid design allows GLiNER to outperform standard NER baselines in both precision and recall, especially in scientific and technical texts. Recent evaluations demonstrate its robustness across multiple domains, including computer science, biomedicine, and social sciences (Zaratiana et al., 2024; Keraghel et al., 2024; Xu et al., 2024). Given its ability to align extracted entities with ontological concepts (e.g., CSO, MeSH), GLiNER is increasingly being adopted for downstream tasks such as KG construction, information extraction, and expert profiling.

We applied GLiNER to the subjects and objects in the triples extracted by OpenIE. For each term, GLiNER assigned relevant topics (it returns a *confidence score* $s \in [0, 1]$ for every detected entity/topic assignment, where s is the model's posterior confidence; higher values indicate greater likelihood of correcteness) based on semantic similarity and pre-

[6] https://github.com/urchade/GLiNER?tab=readme-ov-file

[7] https://github.com/angelosalatino/cso-classifier?tab=readme-ov-file

defined topic categories, enhancing the semantic richness of the data and enabling alignment with KG nodes when possible.

Regarding implementation details, we report the following:

- **Threshold setting:** We employed a threshold of 0.1 to control the sensitivity of the entity recognition process. This threshold balanced recall and precision, ensuring that a comprehensive set of relevant topics was captured. In other words, a lower threshold allowed the model to consider more potential entities, capturing a comprehensive set of relevant topics.
- **Domain-specific labels:** We specified labels such as “tasks”, “methods”, “tools”, and “computer science” to focus the entity recognition process on aspects most relevant to our reviewer recommendation system.

To ensure that the resulting triples contain coherent subjects and objects enriched with meaningful semantic context, we utilized GLiNER for detailed tagging of both subjects and objects output of OpenIE triples. This process involves assigning relevant topics to each term based on semantic similarity and predefined domain-specific categories. As mentioned above, GLiNER is provided with four types of labels: “tasks", “methods", “tools", and “computer science". For example, consider a triple where the subject is “Such models" and the object is “levels on tasks". After applying GLiNER, the subject is categorized under the topic “tools", and the object is categorized under “tasks". This results in enriched triples where both the subject and object carry additional semantic information, facilitating a deeper understanding of their roles within the relationship.

4.2.4 CSO classifier

The CSO Classifier is an unsupervised tool designed to automatically classify research documents according to the CSO (Salatino et al., 2019), a comprehensive taxonomy of research areas in the field of computer science. Leveraging this ontology as a reference framework, the classifier employs a hybrid approach that combines direct string matching with semantic analysis to assign CSO topics using lexical and semantic evidence (Salatino et al., 2019, 2022). The CSO itself is a large-scale, evolving ontology that contains tens of thousands of research topics organized hierarchically and is continuously updated using automated techniques (Salatino et al., 2021). To support this evolution, CSO builds on algorithms such as Klink-2, which infer hierarchical and relatedness links between topics from multiple scholarly sources (Osborne and Motta, 2015). The classifier operates in three modes: (i) *syntactic*, which detects explicit matches of CSO labels in text; (ii) *semantic*, which leverages distributional similarity to identify conceptually related terms; and (iii) *combined*, which integrates both strategies for robust performance (Salatino et al., 2019, 2022). This flexibility allows the CSO Classifier to capture both direct mentions of research fields and latent semantic relations, enabling accurate topic assignment even when terminology varies. It has been successfully applied in large-scale analyses of scientific literature, including monitoring research trends and building scholarly KGs (Salatino et al., 2021), and in editorial/recommendation settings such as Springer Nature’s workflows (Osborne et al., 2016; Salatino et al., 2019). By anchoring extracted concepts to an evolving ontology, the

CSO Classifier provides a consistent and interoperable representation of research domains, which is particularly valuable for tasks requiring longitudinal or cross-venue comparisons.

We utilized the CSO Classifier to extract relevant topics from the outputs generated by OpenIE. The classifier identified topics from the input text and mapped them to the hierarchical structure of the CSO, linking general topics to more specific subfields. This hierarchical mapping not only enriched the semantic representation but also facilitated the integration of the extracted topics into our KGs, enhancing the overall depth and clarity of our constructed semantic structure.

4.2.5 LLM-based approach

In addition to the KG-based approach, we employed LLaMA-3.2 via the Ollama API[8] to extract keywords from the titles, summaries, and full abstracts of both authors' papers and reviewers' publications. For summary generation, we employed the T5-base model[9]. This model, pre-trained on diverse text-to-text tasks, was fine-tuned for domain-specific scientific summarization using the SciTLDR[10]dataset, an established resource containing concise summaries of scientific research articles. While LLaMA-3.2 was employed for keyword extraction, we used the T5-base model for summarization, since T5 is specifically optimized and widely benchmarked for text-to-text tasks such as abstractive summarization. At the time of experimentation, our local LLaMA deployment was not optimized for summarization tasks, and therefore, T5 provided more stable and high-quality summaries.

The fine-tuning process emphasized optimizing the model to generate accurate, domain-relevant, and succinct summaries. By training the model on SciTLDR, which includes short, high-quality summaries tailored for the scientific domain, the summarization aligns with the nuanced demands of reviewer assignment. Summaries were limited to a length of 150 tokens, ensuring clarity and relevance without overwhelming reviewers with excessive detail.

This summarization method produces distilled versions of abstracts that retain the core contributions and findings of the original papers. The summarized abstracts were subsequently integrated into the reviewer assignment system, serving as critical inputs for matching reviewer expertise with paper content. The objective was to generate keywords that effectively align the expertise of reviewers with the topics covered in the submitted papers. This approach enhances the efficiency and precision of reviewer selection by providing concise, informative representations of manuscripts tailored to the scientific context. We utilized the following prompts for the LLaMA-3.2 model:

For the authors' dataset:

"Generate a set of keywords that would help find a reviewer for the following paper `text_type: '{text}'.`"

For the reviewers' dataset:

[8] https://ollama.com/library/llama3.2

[9] https://huggingface.co/docs/transformers/en/model_doc/t5

[10] https://huggingface.co/datasets/allenai/scitldr?row=0

"Propose a set of keywords that define the expertise of the author who wrote a paper with the following `text_type: '{text}'.`*"*

In these templates, the placeholder `{text}` was dynamically replaced with the *title, summary, abstract*, depending on the dataset component. We utilized the *ollama.chat*() function to interact with the model. LLaMA-3.2 produced lists of contextually relevant keywords for each author's paper description or reviewer's profile.

Due to the variability inherent in LLM-generated outputs, we employed NLTK[11] to post-process and clean the extracted text. This approach facilitated robust tokenization, keyword extraction, and normalization of the generated content, ensuring more consistent and reliable analysis.

Following the initial extraction, we implemented a thorough pre-processing phase designed to ensure the accuracy and consistency of the resulting keyword sets. All extracted terms were standardized by converting them to lowercase, removing punctuation, and trimming superfluous whitespace. We then focused on data cleaning, systematically removing common stopwords that did not contribute to semantic content. Lemmatization and stemming techniques were applied to unify words under their canonical forms, further enhancing interpretability. Finally, spell-checking routines were utilized to correct typographical errors, thereby refining the quality and coherence of the extracted keywords. This comprehensive approach ensured that the parsed outputs were both meaningful and readily analyzable for subsequent tasks.

We selected LLaMA-3.2 for keyword extraction due to its availability as an open-source model and its strong performance in language understanding tasks, particularly when deployed locally via the Ollama framework. This setup provided a balance between performance and computational efficiency, enabling large-scale keyword extraction on local machines without requiring paid APIs or external infrastructure. While other LLMs such as GPT-4, Qwen, PHI, or Gemma could also be applied, we prioritized LLaMA-3.2 for reproducibility and accessibility. Benchmarking against alternative models is planned as part of future work.

We represent the LLM-generated keywords using the TF-IDF (Salton and Buckley, 1988) method rather than state-of-the-art transformer embeddings. This choice was made deliberately to keep this component lightweight, transparent, and fully reproducible, and to isolate the contribution of the keyword generation step itself from that of the representation layer. By employing TF-IDF, we ensure that the evaluation focuses on the quality of the keywords produced by the LLM, without introducing additional complexity or confounding effects from deep embedding models

4.3 KG construction

Using the enriched triples, we build the KGs for both authors' papers and reviewers. The graphs represent entities (topics) as nodes and the relationships between them as edges. The nodes and edges are defined as follows:

- **Nodes:** Each node in the graph represents a topic identified by GLiNER and the CSO Classifier from the subjects and objects of the OpenIE triples.

[11] https://www.nltk.org/

- **Edges:** Directed edges connect subject topic nodes to object topic nodes. We considered two scenarios: in the first one, we use the specific relation determined by OpenIE in the triple containing the subject and the object. In the second one, a generic `related_to` relation replaces all specific relations found by OpenIE. Therefore, the number of triples is the same in both scenarios, but in the second one, all have the same relationship.

To enable quantitative comparison between the KGs, we transformed them into vector representations using the Node2Vec algorithm[12]. Node2Vec is a scalable feature learning method that generates vector embeddings for nodes in a graph by simulating biased random walks and applying the skip-gram model (Grover and Leskovec, 2016). It captures both local and global structural information of the graph. Firstly, we configured Node2Vec with specific hyperparameters to optimize embedding quality, the embedding dimension was set to 64 and the length of each random walk was 10 steps with 50 random walks. Secondly, we trained the Node2Vec model on each KG to learn embeddings for all nodes representing topics. This training phase allowed the model to capture both local and global structural features of the graphs, embedding nodes into a continuous vector space where semantically similar topics are positioned closer together. Finally, we computed an overall embedding for each graph by averaging the embeddings of all its constituent nodes. Formally, the graph embedding G is calculated as:

$$G = \frac{1}{N} \sum_{i=1}^{N} v_i,$$

where N denotes the total number of nodes in the graph, and v_i represents the embedding vector of the i-th node. This averaging process yields a fixed-size representation for each graph, facilitating efficient comparison and analysis in subsequent tasks such as similarity measurement and clustering.

We selected Node2Vec over more complex neural graph encoders (e.g., GAT) because our task is fully unsupervised, relies on many small per-document graphs, and benefits from Node2Vec's stable and computationally efficient random-walk embeddings.

4.4 Similarity computation and reviewer recommendation

To recommend suitable reviewers for each paper, we employed several similarity measures that leverage both KG representations and keywords extracted via the LLaMA-3.2 model. For the KG-based approach, we calculated three different similarity metrics: Cosine similarity of graph embeddings, Jaccard similarity of topic sets, and a combined Cosine-Jaccard average similarity. The Cosine similarity measures the orientation similarity between the high-dimensional graph embeddings of papers and reviewers. Given the graph embeddings p for a paper and r for a reviewer, the Cosine similarity is computed as Steck et al. (2024):

$$\text{Cosine Similarity}(p, r) = \frac{p \cdot r}{\|p\| \|r\|}$$

[12] https://snap.stanford.edu/node2vec/

where $p \cdot r$ denotes the dot product of the embedding vectors, and $\|p\|$ and $\|r\|$ are their Euclidean norms.

The Jaccard similarity assesses the overlap between the sets of topics extracted from the KGs. Let T_p and T_r represent the topic sets for the paper and reviewer, respectively, as identified by GLiNER and the CSO classifier. The Jaccard similarity is calculated using (Vedavathi and KM, A.K., 2023):

$$\text{Jaccard Similarity}(T_p, T_r) = \frac{|T_p \cap T_r|}{|T_p \cup T_r|}$$

where $|T_p \cap T_r|$ is the cardinality of the intersection of the topic sets, and $|T_p \cup T_r|$ is the cardinality of their union.

To leverage both structural and thematic similarities, we introduced a combined similarity score by averaging the Cosine and Jaccard similarities: The combined similarity is computed as:

$$\text{Combined Similarity} = \frac{\text{Cosine Similarity} + \text{Jaccard Similarity}}{2}$$

For the LLM-based method, we extracted keywords from titles, summaries, and abstracts using LLaMA-3.2 and represented them numerically using the TF-IDF method. We then computed two primary similarity measures: Jaccard similarity of keyword sets and Cosine similarity of TF-IDF vectors.

The Jaccard similarity between the keyword sets K_p (paper) and K_r (reviewer) is given by:

$$\text{Jaccard Similarity}(K_p, K_r) = \frac{|K_p \cap K_r|}{|K_p \cup K_r|}$$

The Cosine similarity of the TF-IDF vectors v_p (paper) and v_r (reviewer) is calculated as:

$$\text{Cosine Similarity} = \frac{v_p \cdot v_r}{\|v_p\| \, \|v_r\|}$$

By employing these similarity measures, we effectively matched reviewers to authors' papers. This approach ensured that the expertise of the reviewers closely corresponded with the topics of the submitted papers, thereby enhancing the relevance and quality of the peer-review process.

The entire process was implemented in Python[13], utilizing a range of specialized libraries and tools to facilitate graph construction, embedding generation, similarity computations, and text processing. We employed NetworkX[14] for the construction and manipulation of KGs, and Node2Vec was used to generate node and graph embeddings from these graphs. Scikit-learn was instrumental in computing similarity measures, including Cosine similar-

[13] https://www.python.org/

[14] https://networkx.org/documentation/stable/tutorial.html

ity and TF-IDF vectorization, while NumPy[15] assisted in numerical operations and vector calculations to enhance computational efficiency. For NLP tasks such as text parsing and keyword extraction, we utilized NLTK. Additionally, LLaMA-3.2 was integrated via the Ollama API to enable interactions with the LLM and generate contextually relevant keywords. To ensure robustness and consistency in the data processing workflow, methodologies and code implementations were standardized across both the papers' dataset and the reviewers' dataset. Special attention was given to exception handling; in similarity computations, cases where the union of topic sets was empty were carefully managed to avoid division by zero errors. Data structures were optimized for efficiency: graph embeddings were stored as NumPy arrays to facilitate efficient numerical computations and ease of manipulation, while topic sets were represented as Python sets to enable efficient set operations required for calculating Jaccard similarity.

By integrating both KG-based and LLM-based approaches, we enhanced the accuracy and effectiveness of our reviewer recommendation system. The combination of semantic enrichment, structural representation, and advanced language models allowed for a more informed and precise matching between academic papers and potential reviewers.

5 Evaluation

This section delineates the evaluation criteria considered for assessing reviewer recommendation methodologies and subsequently presents the experimental framework employed.

5.1 Evaluation metrics

In the following, we introduce and refine the formal definitions of a selection of widely utilized evaluation metrics. These formulations are adapted for our task of mapping documents to reviewers, wherein a reviewer is deemed *relevant* if they were among the original, empirically assigned reviewers of the corresponding manuscript.

5.1.1 Mean Reciprocal Rank (MRR)

MRR[16] is a position-based metric frequently employed in the evaluation of recommendation and information retrieval systems. Applied to our scenario, the system ranks candidate reviewers for each manuscript, outputting a sorted list in which each reviewer is accompanied by a relevance score. MRR is computed by taking the inverse of the rank of the first relevant reviewer identified for each manuscript, aggregating these values over the entire corpus, and normalizing by the number of manuscripts:

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\mathrm{rank}_i}$$

[15] https://numpy.org/

[16] https://www.evidentlyai.com/ranking-metrics/mean-reciprocal-rank-mrr

Here, $|Q|$ denotes the total number of manuscripts under consideration, and rank_i represents the position of the earliest encountered relevant reviewer for the i-th manuscript. MRR values lie between 0 and 1, with values approaching 1 indicating that relevant reviewers are consistently ranked near the top. While MRR emphasizes the earliest correct retrieval and is thus well-suited to scenarios prioritizing top-ranked matches, its narrow focus limits its capacity to gauge holistic performance. Specifically, it overlooks non-top-ranked relevant reviewers and non-relevant assignments, potentially providing a less comprehensive performance indication (Ali et al., 2021).

5.1.2 Precision at K (P@k)

Precision at k (P@k) quantifies the proportion of top-k recommended reviewers that are relevant. Defined over a fixed cutoff k, it thus evaluates how accurately the system places pertinent reviewers among the first k positions:

$$\mathrm{P@k} = \frac{1}{k} \sum_{i=1}^{k} \begin{cases} 1, & \text{if } r_i \in T \\ 0, & \text{otherwise} \end{cases}$$

In this formulation, r_i denotes the reviewer positioned at rank i, and T represents the set of truly relevant reviewers. P@k values range between 0 and 1, where scores closer to 1 suggest a high density of correct recommendations within the top k positions (Khan et al., 2023).

In our experimental setting, we focus on $k = 3$, reflecting the standard practice of assigning three reviewers per manuscript. Hence, a high P@3 score signifies that the methodology effectively positions pertinent reviewers within these top three choices.

5.1.3 Mean Average Precision (MAP)

MAP[17] extends beyond the first relevant match, integrating precision values at multiple relevant ranks to produce a comprehensive assessment of ranking quality. For a collection of n manuscripts, MAP is defined as:

$$\mathrm{MAP@k} = \frac{1}{n} \sum_{i=1}^{n} \mathrm{AP@k}_i$$

Here, $\mathrm{AP@k}_i$ denotes the average precision at k for the i-th manuscript's ranking list. MAP thus captures the cumulative relevance across all ranks and provides a more holistic perspective than metrics that focus solely on top-ranked items (Zhang et al., 2023).

By considering each relevant reviewer and its respective rank, MAP can yield higher values than P@3 if relevant reviewers appear just beyond the top three positions. For instance, if a manuscript's three relevant reviewers are ranked 1, 3, and 4, then P@3 would be $\frac{2}{3} \approx 0.66$. In contrast, MAP accounts for all relevant reviewers and weights their contri-

[17] https://www.evidentlyai.com/ranking-metrics/mean-average-precision-map

bution by position, leading in this case to an AP of about 0.8. This nuance highlights MAP's ability to capture overall ranking quality rather than focusing only on the very top ranks.

5.2 Experiments

To assess the effectiveness of our proposed reviewer recommendation pipeline, we conducted a series of experiments using the datasets mentioned in Section 4.1. By employing widely recognized information retrieval metrics, namely MAP, MRR, and Precision at k (P@k), detailed in Section 5, we evaluated the quality of the recommended reviewer lists against a ground truth of known reviewers that are placed into the authors' dataset. In our experimental setup, the title, abstract, and summary of each paper were processed using the pipeline described in Section 4.2. For each of these components, similarity scores were computed against the profiles of potential reviewers. These reviewer profiles were derived from their previously published works, from which analogous KGs were constructed and keywords extracted. By evaluating the similarity between the embeddings and keyword profiles of a given paper and those of the potential reviewers, we generated ranked lists of candidate reviewers. We compare the system-generated ranking of reviewers against a set of "ground truth" reviewers. These ground truth assignments were derived from reviewer records within the paper dataset. The experiments, which demonstrate that our integrated KG-based and LLM-based similarity measures significantly improve the alignment between papers and reviewers, challenge the notion that traditional keyword-based methods alone suffice. They verify that topic-enriched knowledge representations enhance recommendation quality. By invoking these methods on real-world data, we observe patterns that suggest improved retrieval of qualified reviewers who align semantically and thematically with the papers under consideration. Our argument speaks to broader issues of automating and refining reviewer selection processes, ultimately facilitating more objective, topic-aligned, and efficient peer review pipelines.

We reused three lexical baselines from our previous work (Bagheri et al., 2024): **TF–IDF Cosine**, **Bag-of-Words Jaccard**, and **Bag-of-Words Dot-Product**. Reviewer profiles are constructed by concatenating the texts of their prior publications, while manuscripts are represented using the matching field (title, abstract, or summary). For each baseline, we report two aggregation strategies across a reviewer's publications *Mean* and *Max* similarity scores. Performance is evaluated using MRR, MAP, and P@3.

5.3 Workload and fairness metrics

To evaluate the fairness and efficiency of reviewer assignments, we define several workload-related metrics. Denote by $w_{r,c}$ the number of papers assigned to reviewer r in conference c. All metrics are computed on a per-conference basis, considering only those reviewers who have received at least one assignment in the respective conference.

Per-conference average workload For each conference c, define R_c as the set of reviewers with at least one assignment in that conference. The average workload (Kobren et al., 2019) for conference c is then given by:

$$\bar{w}_c = \frac{1}{|R_c|} \sum_{r \in R_c} w_{r,c}$$

Macro-average workload To summarize workloads uniformly across all conferences, we compute the unweighted average (Sokolova and Lapalme, 2009) of the per-conference means:

$$\text{MacroAvg} = \frac{1}{|\mathcal{C}'|} \sum_{c \in \mathcal{C}'} \bar{w}_c$$

where $\mathcal{C}'$ denotes the set of conferences in which at least one assignment was made. This metric assigns equal weight to each conference, independent of its size.

Pooled mean workload To reflect the overall average number of assignments per reviewer–conference pair, we compute the pooled mean (Sokolova and Lapalme, 2009):

$$\text{Pooled} = \frac{\sum_c \sum_{r \in R_c} w_{r,c}}{\sum_c |R_c|}$$

This measure aggregates all reviewer assignments across all conferences and divides by the total number of reviewer–conference pairs with at least one assignment. Unlike the macro-average, it weights each conference proportionally to its number of active reviewers.

Gini coefficient (Workload Inequality). To assess the fairness of the workload distribution, we employ the Gini coefficient G (Ceriani and Verme, 2012), a standard metric for measuring inequality. Applied to the multiset $\{w_{r,c}\}$, it is computed as:

$$G = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} |x_i - x_j|}{2n^2 \bar{x}}$$

Here, x_i represents the individual workloads, $\bar{x}$ is their mean, and n is the number of reviewer–conference pairs with nonzero load. A Gini coefficient of 0 indicates perfect equality, while values closer to 1 denote greater inequality[18].

6 Results

Below, we present the results of our evaluation organized into five tables corresponding to different experimental conditions and similarity measures. Tables 1 and 2 report performance metrics for KG-based approaches using "relation" and `related_to` edges, respectively, assessed by Cosine similarity, Jaccard similarity, and their average (Cosine–Jaccard average similarity). Tables 3, 4, and 5 summarize the outcomes for LLM-based methods compared against baseline similarity approaches on abstract, summary, and title representa-

[18] Equivalently, the Gini coefficient is twice the area between the Lorenz curve and the line of perfect equality.

Table 1 Reviewer matching using KGs and relation edge

Evaluation Metrics	Open IE relation	CSO-Classifier relation	GLiNER relation
Cosine Similarity			
MRR	0.5714	**0.7619**	0.5277
P@3	0.3284	**0.3445**	0.3300
MAP	0.2182	**0.2539**	0.1851
Jaccard Similarity			
MRR	0.7112	**0.7424**	0.6354
P@3	0.3326	**0.3471**	0.3326
MAP	0.2370	**0.2475**	0.2274
Cosine-Jaccard Average Similarity			
MRR	0.6721	**0.7857**	0.5833
P@3	0.3401	**0.3920**	0.3313
MAP	0.2395	**0.2936**	0.2037

Best values are shown in bold

Table 2 Reviewer matching using KGs and `related_to` edge

Evaluation Metrics	Open IE `related_to`	CSO-Classifier `related_to`	GLiNER `related_to`
Cosine Similarity			
MRR	0.5625	**0.6711**	0.6666
P@3	0.3750	**0.3888**	0.3379
MAP	0.2743	**0.2592**	0.2460
Jaccard Similarity			
MRR	0.5952	0.5333	**0.6674**
P@3	0.3333	0.3239	**0.4285**
MAP	0.2301	0.1778	**0.2857**
Cosine-Jaccard Average Similarity			
MRR	0.6212	**0.7142**	0.3571
P@3	0.3570	**0.3809**	0.2955
MAP	0.2373	**0.2698**	0.1190

Best values are shown in bold

Table 3 Reviewer matching using the abstracts of papers

	Abstract of Papers					
	LLM Method			MSAAR		
	Cosine Similarity	Jaccard Similarity	Cosine-Jaccard Average Similarity	Cosine Similarity	Jaccard Similarity	Dot Product Similarity
MRR	0.6584	**0.7112**	0.6364	0.6958	0.5737	0.6263
P@3	**0.3703**	0.3450	0.3674	0.3259	0.3125	0.3071
MAP	0.2551	0.2643	0.2455	0.3995	**0.5881**	0.3737

Best values are shown in bold

Table 4 Reviewer matching using the summaries of abstracts

	Summary of abstracts					
	LLM Method			MSAAR		
	Cosine Similarity	Jaccard Similarity	Cosine-Jaccard Average Similarity	Cosine Similarity	Jaccard Similarity	Dot Product Similarity
MRR	**0.6534**	0.6501	0.6503	0.238	0.3095	0.5476
P@3	**0.3674**	0.3557	0.3550	0.0845	0.1673	0.3202
MAP	0.2522	0.2462	0.2415	0.1369	0.1964	**0.3452**

Best values are shown in bold

Table 5 Reviewer matching using the titles of papers.

	Titles of papers					
	LLM Method			MSAAR		
	Cosine Similarity	Jaccard Similarity	Cosine-Jaccard Average Similarity	Cosine Similarity	Jaccard Similarity	Dot Product Similarity
MRR	0.6715	**0.6848**	0.6401	0.3669	0.07692	0.6111
P@3	0.3627	**0.3645**	0.3613	0.1409	0.0192	0.1809
MAP	0.2570	0.2621	0.2440	0.2273	0.0384	**0.3611**

Best values are shown in bold

tions of research papers. Our analysis addresses how semantic enrichment, domain-specific ontologies, and advanced keyword extraction affect reviewer recommendation accuracy.

As illustrated in Table 1, which focuses on "relation" edges, the integration of the CSO classifier consistently enhances performance. For instance, under Cosine-Jaccard Average Similarity, CSO classifier relations yield an MRR of 0.7857, surpassing both OpenIE relations (0.6721) and GLiNER relations (0.5833). Similar trends persist across the Jaccard similarity and Cosine similarity measures, indicating that domain-specific topic enrichment bolsters thematic fidelity between papers and reviewers.

Table 2 presents outcomes for the `related_to` condition and confirms the positive impact of semantic enrichment. While OpenIE `related_to` achieves moderate MRR values (e.g., 0.5625 under Cosine Similarity), CSO classifier and GLiNER both improve rankings and precision. Notably, the CSO classifier `related_to` configuration attains a higher MRR (0.7142) under the Cosine-Jaccard Average Similarity metric compared to OpenIE (0.6212) and GLiNER (0.3571). These gains indicate that leveraging hierarchical ontology structures and advanced entity recognition techniques refines the semantic representation, improving reviewer-paper alignment. Our new findings underscore that the inclusion of domain-specific ontologies and refined entity extraction methods, which was not fully explored in our previous work, leads to a deeper and more context-aware characterization of research topics.

Based on the comparative results in Tables 1 and 2, we identify the optimal configuration for our KG-based approach. The strongest overall performance is obtained when constructing KGs with the CSO Classifier while preserving the original relation edges and computing similarity using the Cosine-Jaccard average. This setup achieves the highest MRR (0.7857)

and competitive MAP and P@3 values in Table 1. By contrast, collapsing all edges into the generic `related_to` relation improves robustness in some scenarios but underperforms in terms of early precision (e.g., MRR = 0.7142 in Table 2). Accordingly, we recommend CSO + relation edges + Cosine–Jaccard average as the default configuration for reviewer assignment.

Tables 3, 4, and 5 compare LLM-based approaches to the baseline methods (Maximum of Similarities across all Reviewers' Articles (MSAAR)) described in our previous paper (Bagheri et al., 2024), using keywords derived from abstracts, summaries, and titles. For abstracts (Table 3), LLM-based methods yield competitive MRR and P@3 scores. For example, under Jaccard Similarity, the LLM-based approach reaches an MRR of 0.7112, slightly outperforming the baseline Cosine similarity MRR of 0.6958. Although the baseline still shows a higher MAP (0.3995 compared to 0.2643 for the LLM-based approach), the LLM-driven keyword extraction ensures more robust thematic capture than our earlier methodologies.

Similar patterns emerge with summaries (Table 4). The LLM-based Cosine similarity approach achieves an MRR of 0.6534, markedly higher than the baseline Cosine similarity (0.238) reported in our previous work. This demonstrates that even with shorter textual inputs, leveraging advanced LLM-generated keywords and similarity measures results in improved alignment between manuscripts and reviewers.

Turning to titles (Table 5), the LLM-based Cosine similarity (MRR=0.6715) again surpasses the baseline Cosine similarity (MRR=0.3669), highlighting the ability of LLMs to extract semantically rich keywords from minimal textual content. Moreover, Jaccard similarity with LLM-based keywords (MRR=0.6848) also exceeds the baseline metrics, confirming that these enhanced keywords offer substantial improvements over traditional, lexically driven approaches.

It is important to note that TF-IDF and Node2Vec represent two fundamentally different perspectives on expertise. TF-IDF applied to LLM-generated keywords emphasizes lexical salience and transparency, whereas Node2Vec embeddings capture structural and relational information from KGs. Their comparison in Tables 1–5 is therefore not redundant, but rather highlights complementary signals: keyword-driven lexical evidence versus topology-driven structural embeddings. This distinction explains why performance differences arise across settings, with TF-IDF excelling in lightweight lexical matching and Node2Vec offering richer relational modeling.

To assess robustness, we ran paired *t*-tests on per-paper Average Precision (AP), the basis of MAP. Under our conservative protocol, the CSO Classifier with relation edges (Table 1) consistently attains the best or co-best MRR/P@3/MAP across similarities; however, the observed deltas against OpenIE/GLiNER do not reach $p<0.05$ on the paired overlaps. Effect-size estimates (Cohen's d in the range 0.15–0.25) indicate only small practical differences. When collapsing edges to `related_to` (Table 2), changes likewise remain non-significant, though the cosine–Jaccard average shows small, positive trends.

For the LLM-based variants (Tables 3-5), title and summary settings exhibit small numerical gains for the cosine–Jaccard average over cosine, but these differences again fall short of $p<0.05$; abstract-level comparisons are similarly indistinguishable under paired tests. Effect sizes here are negligible ($d<0.10$), reinforcing the statistical comparability of LLM-based keywording and cosine baselines.

Overall, our statistical analysis finds no broad advantages ($p<0.05$) under the current protocol and sample size. This may reflect limited statistical power for detecting small improvements on our dataset. Accordingly, we report effect-size deltas with 95% confidence intervals and interpret non-significant differences as expected variation. In practice, the methods are statistically comparable, with consistent numerical advantages for CSO with relation edges and small positive trends for `related_to` with cosine–Jaccard. Larger-scale studies may be needed to establish whether these trends generalize.

To evaluate the practical feasibility of our pipeline, we measured the average runtime of each processing stage on a workstation equipped with an NVIDIA RTX 3090 GPU, AMD Ryzen 9 CPU, and 128 GB RAM. KG construction for a single abstract required on average 0.8 seconds, while generating random-walk embeddings with Node2Vec took 3.1 seconds per graph. Similarity computation between a paper and all candidate reviewers was lightweight, averaging 0.5 seconds. For the LLM-based component, keyword extraction with LLaMA-3.2 required approximately 4.2 seconds per abstract when executed locally via Ollama, whereas summarization with the T5-base model averaged 1.6 seconds per abstract. Overall, processing a paper end-to-end, including KG construction, embedding, and similarity scoring, took less than 5 seconds, and keyword-based matching less than 6 seconds, making the system practical for batch processing of hundreds of submissions within minutes. Since KG construction and reviewer graph embeddings can be precomputed offline, the online reviewer assignment stage remains efficient and scalable for real-world peer-review workflows.

6.1 Illustrative case-based analysis

To complement the numerical metrics reported in the previous subsections, we provide case-based examples that illustrate how our pipeline behaves in practice. These examples explain why integrating OpenIE, CSO Classifier, and GLiNER, with different edge configurations in the KGs, leads to the observed gains over baselines.

Case A — CSO+relation and early precision gains Abstract-derived triples anchor paper concepts that the CSO Classifier maps to ontology nodes (e.g., `knowledge-graphs`, `relation-extraction`). Random-walk embeddings then place papers and reviewers close when they co-occupy semantically adjacent CSO regions, even if the wording differs. Preserving the original OpenIE relation edge retains informative structure (e.g., `task-used-in-method`, `model-improves-on-metric`). This structure lifts the correct reviewer to rank 1 more often than OpenIE or GLiNER KGs, in line with Table 1: CSO+relation achieves the best overall early-precision metrics (MRR $= \mathbf{0.7857}$, P@3 $= \mathbf{0.3920}$, MAP $= \mathbf{0.2936}$) under Cosine–Jaccard averaging.

Case B — effectiveness of `related_to` with GLiNER and Jaccard Collapsing edges to `related_to` reduces brittleness from noisy predicates (common with OpenIE on scientific prose) but makes the graph denser. GLiNER's fine-grained labels ("tasks", "methods", "tools") constrain node types, which pairs naturally with set-overlap measures (Jaccard). With `related_to` edges, GLiNER+Jaccard increases hit coverage within top-k even if top-1 is not always optimal: Table 2 shows the best P@3 ($\mathbf{0.4285}$) and MAP ($\mathbf{0.2857}$) for GLiNER under Jaccard, indicating more correct reviewers retrieved near the top of the list.

Case C — LLM-based matching for titles and summaries Because OpenIE on titles/summaries is sparse, our KGs are **abstract-driven**; for title/summary scenarios, we therefore rely on **LLM-derived keywords** (noisy-signal denoising) rather than KG construction. The LLM condenses the intent and expands synonyms, which lexical baselines often miss. LLM-based matching improves early precision over baselines (Table 5): best LLM's MRR $= \mathbf{0.6848}$ and P@3 $= \mathbf{0.3645}$ vs. baseline best P@3 $= 0.1809$. Baselines may yield higher MAP in some cases, but early ranks, most critical for assignments, favor LLM-based methods. The LLM summary denoises long abstracts, focusing on salient topics; LLM's MRR $= \mathbf{0.6534}$, P@3 $= \mathbf{0.3674}$ outperform baselines' early precision (Table 4), while a baseline achieves higher MAP, reflecting recovery of additional relevant reviewers deeper in the list.

Case D — failure patterns and mitigation OpenIE may produce generic predicates (e.g., "achieves", "based-on") or boundary errors, creating low-information or spurious edges. In `relation`-edge graphs, this can mislead proximity; in `related_to`-edge graphs, it can over-connect hubs. We prune malformed/low-support triples, use CSO/GLiNER to regularize node types, and evaluate both edge settings: `related_to` improves robustness (GLiNER+Jaccard in Table 2), while relation retains specificity (CSO in Table 1). Residual rank inversions typically trace back to a handful of noisy triples or ambiguous acronyms.

- Summary of findings **CSO + relation edges** maximize early precision across metrics (Table 1), explaining the observed MRR/P@3 gains.
- **GLiNER** + `related_to` + **Jaccard** increases top-k coverage (Table 2), reflecting better MAP/P@3 under coarser but robust connectivity.
- **LLM on titles/summaries** improves early ranking where OpenIE-based KGs are impractical (Tables 5 and 4); baselines can win MAP in some cases, signaling stronger depth but weaker top-rank precision.

6.2 Workload and fairness (Per-Conference)

In addition to ranking performance, we assessed the fairness and realism of the reviewer workloads produced by each method, aggregating results on a per-conference basis. For the LLM-based approach using titles, summaries, and abstracts, the average number of reviews per reviewer computed as the unweighted macro-average across all conferences is 2.27. When pooling all reviewer conference pairs (i.e., weighting conferences by size), the average workload increases to 3.00. The Gini coefficient, which quantifies the inequality in the distribution of review assignments, is 0.243, indicating moderate imbalance.

Comparable statistics are observed for the KG variants. Specifically:

- **OpenIE-based KG:** macro-average workload of 2.18, pooled mean of 2.86, and Gini coefficient of 0.249.
- **GLiNER-based KG:** macro-average workload of 2.02, pooled mean of 2.31, and Gini coefficient of 0.243.
- **CSO-based KG:** macro-average workload of 2.06, pooled mean of 2.49, and Gini coefficient of 0.225.

Overall, all methods yield reviewer workloads in a practical range approximately 2 to 3 papers per reviewer—with Gini values between 0.22 and 0.25, reflecting low to moderate levels of inequality in assignment distribution.

7 Conclusions and future work

In this paper, we introduced a novel reviewer assignment pipeline utilizing KGs and LLMs to address inefficiencies in traditional peer review matching. The framework constructs KGs for papers and reviewers using three knowledge extraction methods: Open IE, CSO classifier, and GLiNER on paper abstracts. On the other hand, LLMs generate targeted keywords to refine expertise representation, identifying reviewers for candidate papers and summarizing expertise from reviewers' publications. Similarity between paper-reviewer KGs and extracted keywords by LLM are computed via Cosine, Jaccard, and hybrid metrics, with performance evaluated against ground-truth data using MRR, MAP, and P@K. When using KGs constructed from Open IE, CSO classifier, and GLiNER outputs, we observe that the CSO classifier relation consistently delivers higher MRR and MAP with 0.7857 and 0.2936 values, respectively, especially when paired with the Cosine-Jaccard average similarity measure. On the other hand, on comparing textual representations, the LLM-based method outperforms the baseline particularly well on summaries of abstracts and titles. For instance, using summaries, the LLM method consistently yields higher MRR and P@3 values than the baseline across Cosine, Jaccard, and their averaged similarity metrics. Similarly, for titles, our approach exhibits strong performance, with the LLM method attaining notably higher scores in Cosine and Jaccard similarities compared to the baseline. Although the baseline method shows competitive performance on full abstracts in some metrics, our approach offers a more balanced performance overall, especially in ranking and precision at the top positions. In conclusion, these results validate that leveraging LLM-based keyword extraction and KGs representations can effectively bridge the semantic gap between papers and reviewer profiles. The integration of multiple similarity measures, particularly the hybrid Cosine-Jaccard metric, further refines this matching process, yielding enhanced reviewer assignment accuracy.

Our current evaluation is limited to computer science data; generalizability to other fields remains open. The pipeline depends on OpenIE quality, and we have not yet explored advanced neural graph encoders. Model diversity for summarization and keywording is also limited.

Future work will extend evaluation beyond computer science by adding cross-disciplinary datasets (e.g., biomed, materials). We will run an ablation over three configurations: (i) *KG-only* (OpenIE + CSO/GLiNER with Node2Vec), (ii) *LLM-only* (LLM keywords with TF–IDF similarity), and (iii) *KG+LLM* (late-fusion of the two similarity scores). We will further include (a) a *cold-start* protocol that truncates reviewer histories and (b) an *emerging-topics* protocol using a temporal split (train $\leq t$, test $> t$ on topics first appearing after t), and report both accuracy (MRR/MAP/P@3) and fairness metrics (coverage, load balance).

Future work will also explore stronger baselines such as TPMS, BERT-based retrieval, and PeerReview4All assignment. We will also investigate fairness- and COI-aware constraints to ensure balanced and unbiased reviewer allocation. In this study, KGs and LLMs were applied as separate, comparable tracks; future work will explore hybrid models that

integrate them more tightly (late- or joint-fusion). Finally, we plan to benchmark alternative LLMs (e.g., GPT-4, Qwen, Phi, Gemma) for the keyword extraction component, systematically assessing their impact on reviewer assignment performance. In future work, we also plan to benchmark alternative LLMs not only for keyword extraction but also for scientific summarization, in order to assess their impact across both components of the pipeline.

Author Contributions F.B. developed the code, conducted the experiments, and drafted the manuscript. D.B. conceived the methodology, provided supervision, and revised the manuscript. D.R.R. supervised the research and contributed to manuscript revision. All authors reviewed and approved the final version of the manuscript.

Funding Open access funding provided by Università degli Studi di Cagliari within the CRUI-CARE Agreement. Funding not applicable.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Conflicts of Interest The authors declare that they have no conflict of interest.

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Adebiyi, A.A., Ogunleye, O.M., Adebiyi, M., et al. (2019). A comparative analysis of tf-idf, lsi and lda in semantic information retrieval approach for paper-reviewer assignment. *Journal Of Engineering And Applied Sciences, 14*(10), 3378–3382. https://doi.org/10.36478/jeasci.2019.3378.3382

Aksoy, M., Yanik, S., & Amasyali, M. F. (2023). Reviewer assignment problem: A systematic review of the literature. *Journal Of Artificial Intelligence Research, 76*, 761–827. https://doi.org/10.1613/jair.1.14318

Ali, Z., Ullah, I., Khan, A., Jan, U., et al. (2021). An overview and evaluation of citation recommendation models. *Scientometrics, 126*, 4083–4119. https://doi.org/10.1007/s11192-021-03909-y

Angeli, G., Premkumar, M.J.J., Manning, C.D. (2015). Leveraging linguistic structure for open domain information extraction. In: Proceedings Of The 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (Volume 1: Long Papers). pp. 344–354. https://doi.org/10.3115/v1/P15-1034

Bagheri, F., Buscaldi, D., Recupero, D.R. (2024). A study on content-based reviewer assignment in the semantic web and computer science domains. In: LKE 2024, language and knowledge engineering. https://doi.org/10.13053/cys-28-4-5299

Bhaisare, B., Bharati, R. (2024). Advancing peer review integrity: Automated reviewer assignment techniques with a focus on deep learning applications. In: International conference on computation of artificial intelligence & machine learning. pp. 312–327. Springer. https://doi.org/10.1007/978-3-031-71481-8_25

Cagliero, L., Garza, P., Pasini, A., et al. (2018). Additional reviewer assignment by means of weighted association rules. *IEEE Transactions On Emerging Topics In Computing, 9*(1), 329–341. https://doi.org/10.1109/TETC.2018.2861214

Cai, X., Xiao, M., Ning, Z., et al. (2023). Resolving the imbalance issue in hierarchical disciplinary topic inference via llm-based data augmentation. In: 2023 IEEE international conference on data mining workshops (ICDMW), pp. 1424–1429. IEEE. https://doi.org/10.1109/ICDMW60847.2023.00181

Ceriani, L., & Verme, P. (2012). The origins of the gini index: extracts from variabilità e mutabilità (1912) by corrado gini. *The Journal Of Economic Inequality, 10*(3), 421–443. https://doi.org/10.1007/s10888-011-9188-x

Charlin, L., Zemel, R.S. (2013) The toronto paper matching system: An automated paper–reviewer assignment system. In: ICML workshop on peer reviewing and publishing models (ICML PeerReview). https://www.cs.toronto.edu/ lcharlin/papers/tpms.pdf

Duan, Z., Tan, S., Zhao, S., et al. (2019). Reviewer assignment based on sentence pair modeling. *Neurocomputing, 366*, 97–108. https://doi.org/10.1016/j.neucom.2019.06.074

Giarelis, N., Mastrokostas, C., Karacapilidis, N. (2024). A unified LLM-KG framework to assist fact-checking in public deliberation. In: Hautli-Janisz, A., Lapesa, G., Anastasiou, L., Gold, V., Liddo, A.D., Reed, C. (eds.) Proceedings of the First Workshop on Language-driven Deliberation Technology (DELITE) @ LREC-COLING 2024. pp. 13–19. ELRA and ICCL, Torino, Italia. https://aclanthology.org/2024.delite-1.2/

Grover, A., Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In: Proceedings Of The 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 855–864. https://doi.org/10.1145/2939672.2939754

Keraghel, I., Morbieu, S., Nadif, M. (2024). Recent advances in named entity recognition: A comprehensive survey and comparative study. https://doi.org/10.48550/arXiv.2401.10825

Khan, F., Al Rawajbeh, M., Ramasamy, L. K., et al. (2023). Context-aware and click session-based graph pattern mining with recommendations for smart ems through ai. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3285552

Kobren, A., Saha, B., McCallum, A. (2019). Paper matching with local fairness constraints. In: Proceedings Of The 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 1247–1257. https://doi.org/10.1145/3292500.3330899

Kusumawardani, R.P., Khairunnisa, S.O. (2018). Author-topic modelling for reviewer assignment of scientific papers in bahasa indonesia. In: 2018 International conference on asian language processing (IALP). pp. 351–356. IEEE. https://doi.org/10.1109/IALP.2018.8629124

Li, K., Cao, Z., Qu, D. (2017). Fair reviewer assignment considering academic social network. In: Web And Big Data: First International Joint Conference, APWeb-WAIM 2017, Beijing, China, July 7–9, 2017, Proceedings, Part I 1. pp. 362–376. Springer. https://doi.org/10.1007/978-3-319-63579-8_28

Li, C., Shi, Y., Luo, Y., et al. (2025). Rise of the community champions: From reviewer crunch to community power. ArXiv Preprint. https://doi.org/10.48550/arXiv.2503.18336

Madzík, P., & Falát, L. (2022). State-of-the-art on analytic hierarchy process in the last 40 years: Literature review based on latent dirichlet allocation topic modelling. *PLoS One, 17*(5), Article e0268777. https://doi.org/10.1371/journal.pone.0268777

Marchesin, S., Silvello, G., Alonso, O. (2025) Large language models and data quality for knowledge graphs. *Information Processing & Management, 62*(6), 104281. https://doi.org/10.1016/j.ipm.2025.104281.

Mariotti, L., Guidetti, V., Mandreoli, F., et al. (2024). Combining large language models with enterprise knowledge graphs: a perspective on enhanced natural language understanding. *Frontiers In Artificial Intelligence, 7*, 1460065. https://doi.org/10.3389/frai.2024.1460065

Misale, M., Vanwari, P. (2017). A survey on recommendation system for technical paper reviewer assignment. In: 2017 International conference of electronics, communication and aerospace technology (ICECA). vol. 2, pp. 329–331. IEEE. https://doi.org/10.1109/ICECA.2017.8212826

Mittal, K., Jain, A., Vaisla, K.S. (2019). Understanding reviewer assignment problem and its issues and challenges. In: 2019 4th International Conference On Internet Of Things: Smart Innovation And Usages (IoT-SIU). pp. 1–6. IEEE. https://doi.org/10.1109/IoT-SIU.2019.8777727

Mittal, K., Jain, A., Vaisla, K.S., et al. (2020). A novel method for reviewer assignment problem based on reviewers' profile and fuzzy graph connectivity measure. In: 2020 International conference on intelligent engineering and management (ICIEM). pp. 386–391. IEEE https://doi.org/10.1109/ICIEM48762.2020.9160042

Nugroho, A.S., Saikhu, A., Anggraini, R.N.E., et al. (2023). Development of reviewer assignment method with latent dirichlet allocation and link prediction to avoid conflict of interest. Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi) *7*(4),837–844 .https://doi.org/10.29207/resti.v7i4.4900

OpenReview (2023). Openreview expertise: Paper–reviewer affinity modeling. https://github.com/openreview/openreview-expertise, gitHub repository

OpenReview: Openreview matcher (2019). Optimal paper–reviewer matching with constraints. https://github.com/openreview/openreview-matcher, gitHub repository

OpenReview (2025) Paper matching and assignment. https://docs.openreview.net/how-to-guides/paper-matching-and-assignment, documentation

Osborne, F., Motta, E. (2015). Klink-2: integrating multiple web sources to generate semantic topic networks. In: International Semantic Web Conference. pp. 408–424. Springer. https://doi.org/10.1007/978-3-319-25007-6_24

Osborne, F., Salatino, A., Birukou, A., et al. (2016). Automatic classification of springer nature proceedings with smart topic miner. In: The Semantic Web–ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part II 15. pp. 383–399. Springer. https://doi.org/10.1007/978-3-319-46547-0_33

Pan, S., Luo, L., Wang, Y., et al. (2024). Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions On Knowledge And Data Engineering, 36*(7), 3580–3599. https://doi.org/10.1109/TKDE.2024.3352100

Payan, J. (2022). Fair allocation problems in reviewer assignment. In: Proceedings of the 21st international conference on autonomous agents and multiagent systems. p. 1857–1859. AAMAS '22, international foundation for autonomous agents and multiagent systems, Richland, SC

Peng, H., Hu, H., Wang, K., et al. (2017). Time-aware and topic-based reviewer assignment. In: Database Systems For Advanced Applications: DASFAA 2017 International Workshops: BDMS, BDQM, SeCoP, And DMMOOC, Suzhou, China, March 27-30, 2017, Proceedings 22. pp. 145–157. Springer. https://doi.org/10.1007/978-3-319-55705-2_11

Qader, W.A., Ameen, M.M., Ahmed, B.I. (2019). An overview of bag of words; importance, implementation, applications, and challenges. In: 2019 International Engineering Conference (IEC). pp. 200–204. IEEE. https://doi.org/10.1109/IEC47844.2019.8950616

Reimers, N., Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. ArXiv Preprint. https://doi.org/10.48550/arXiv.1908.10084

Rordorf, D., Käser, J., Crego, A., et al. (2023). A hybrid intelligent approach combining machine learning and a knowledge graph to support academic journal publishers addressing the reviewer assignment problem (RAP). In: Martin, A., Fill, H., Gerber, A., Hinkelmann, K., Lenat, D., Stolle, R., van Harmelen, F. (eds.) Proceedings of the AAAI 2023 Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering (AAAI-MAKE 2023), Hyatt Regency, San Francisco Airport, California, USA, March 27-29, 2023. CEUR Workshop Proceedings, vol. 3433. CEUR-WS.org. https://ceur-ws.org/Vol-3433/paper15.pdf

Salatino, A.A., Mannocci, A., Osborne, F. (2021). Detection, Analysis, and Prediction of Research Topics with Scientific Knowledge Graphs, pp. 225–252. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-86668-6_11

Salatino, A.A., Osborne, F., Birukou, A., et al. (2019). Improving editorial workflow and metadata quality at springer nature. In: International semantic web conference. pp. 507–525. Springer. https://doi.org/10.1007/978-3-030-30796-7_31

Salatino, A., Osborne, F., Motta, E. (2022). Cso classifier 3.0: a scalable unsupervised method for classifying documents in terms of research topics. *International Journal On Digital Libraries, 23*(1),91–110. https://doi.org/10.1007/s00799-021-00305-y

Salatino, A.A., Osborne, F., Thanapalasingam, T., et al. (2019). The cso classifier: Ontology-driven detection of research topics in scholarly articles. In: Digital libraries for open knowledge: 23rd international conference on theory and practice of digital libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings 23. pp. 296–311. Springer. https://doi.org/10.48550/arXiv.2104.00948

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513–523. https://doi.org/10.1016/0306-4573(88)90021-0

Sarzynska-Wawer, J., Wawer, A., Pawlak, A., et al. (2021). Detecting formal thought disorder by deep contextualized word representations. *Psychiatry Research, 304*, Article 114135. https://doi.org/10.1016/j.psychres.2021.114135

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management, 45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Steck, H., Ekanadham, C., Kallus, N. (2024). Is cosine-similarity of embeddings really about similarity? ArXiv Preprint. https://doi.org/10.48550/arXiv.2403.05440

Stelmakh, I., Shah, N.B., Singh, A. (2019). Peerreview4all: Fair and accurate reviewer assignment in peer review. In: Algorithmic learning theory. pp. 828–856. PMLR. https://doi.org/10.48550/arXiv.1806.06237

Sun, Y., Huang, Q., Tang, Y., et al. (2024). A general framework for producing interpretable semantic text embeddings. ArXiv Preprint. https://doi.org/10.48550/arXiv.2410.03435

Tong, W., Chu, X., Li, Z., et al. (2024). Generative adversarial meta-learning knowledge graph completion for large-scale complex knowledge graphs. *Journal of Intelligent Information Systems, 62*, 1685–1701. https://doi.org/10.1007/s10844-024-00860-1

Vedavathi, N., & KM, A.K. (2023). E-learning course recommendation based on sentiment analysis using hybrid elman similarity. *Knowledge-Based Systems, 259*, 110086 . https://doi.org/10.1016/j.knosys.2022.110086

Wang, L., Gan, Y., Wang, X., et al. (2025). Textual and structural dual enhancement for knowledge graph completion with large language models. *Journal of Intelligent Information Systems, 63*, 1625–1643. https://doi.org/10.1007/s10844-025-00953-5

Xiao, M., Qiao, Z., Fu, Y., et al. (2022). Who should review your proposal? interdisciplinary topic path detection for research proposals. ArXiv Preprint. https://doi.org/10.48550/arXiv.2203.10922,

Xiao, M., Qiao, Z., Fu, Y., Dong, H., Du, Y., Wang, P., Xiong, H., & Zhou, Y. (2023). Hierarchical interdisciplinary topic detection model for research proposal classification. *IEEE Transactions On Knowledge And Data Engineering, 35*(9), 9685–9699. https://doi.org/10.1109/TKDE.2023.3248608

Xu, D., Chen, W., Peng, W., et al. (2024). Large language models for generative information extraction: A survey. *Frontiers Of Computer Science, 18*(6), Article 186357. https://doi.org/10.1007/s11704-024-40555-y

Yong, Y., Yao, Z., Zhao, Y. (2021). A framework for reviewer recommendation based on knowledge graph and rules matching. In: 2021 IEEE international conference on information communication and software engineering (ICICSE). pp. 199–203. https://doi.org/10.1109/ICICSE52190.2021.9404099

Zaratiana, U., Tomeh, N., Holat, P., et al. (2024). GLiNER: Generalist model for named entity recognition using bidirectional transformer. In: Duh, K., Gomez, H., Bethard, S. (eds.) Proceedings Of The 2024 conference of the north american chapter of the association for computational linguistics: human language technologies (Volume 1: Long Papers). pp. 5364–5376. Association for Computational Linguistics, Mexico City, Mexico. https://doi.org/10.18653/v1/2024.naacl-long.300

Zhang, P., Fu, P., Chen, K., et al. (2024). A novel paper-reviewer recommendation method based on a semantics and correlation fusion model. In: Proceedings of the international conference on computing, machine learning and data science. pp. 1–6. https://doi.org/10.1145/3661725.3661748

Zhang, Y., Shen, Y., Kang, S., et al. (2025). Chain-of-factors paper-reviewer matching. In: Proceedings Of The ACM On web conference 2025. pp. 1901–1910. WWW '25 https://doi.org/10.1145/3696410.3714708

Zhang, T., Zhang, Y., Xin, M., et al. (2023). A light-weight network for small insulator and defect detection using uav imaging based on improved yolov5. *Sensors, 23*(11), 5249. https://doi.org/10.3390/s23115249

Zhao, X., & Zhang, Y. (2022). Reviewer assignment algorithms for peer review automation: A survey. *Information Processing & Management, 59*(5), 103028. https://doi.org/10.1016/j.ipm.2022.103028

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.