DEMONSTRATION

ReviewHQ: An API-Based System for Reviewer Assignment and Quality Control in Research Conferences

GUIDO ZUCCON, The University of Queensland, Brisbane, QLD, Australia

Open Access Support provided by:

The University of Queensland

Published: 13 July 2025

Citation in BibTeX format

SIGIR '25: The 48th International ACM SIGIR Conference on Research and Development in Information Retrieval
July 13 - 18, 2025
Padua, Italy

Conference Sponsors:
SIGIR

SIGIR '25: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (July 2025)
https://doi.org/10.1145/3726302.3730139
ISBN: 9798400715921

ReviewHQ: An API-Based System for Reviewer Assignment and Quality Control in Research Conferences

Guido Zuccon
g.zuccon@uq.edu.au
The University of Queensland
St Lucia, Queensland, Australia

Abstract

Managing the review process for large-scale academic conferences poses significant challenges in effectively matching papers to the right reviewers, detecting conflicts of interest, ensuring review quality, and addressing potential ethical issues such as dual submissions. In this demonstration paper, we introduce ReviewHQ, an API-based system designed to streamline conference management.

For matching reviewers to publications, ReviewHQ constructs expertise profiles for potential reviewers by mining their publication records and representing both papers and reviewers as dense vectors. It then formulates paper assignment as a constrained optimization problem, leveraging dense vector similarity scores and other reviewer's and submission's features to produce high-quality assignments. Beyond its core functionality, ReviewHQ identifies conflicts of interest by analyzing co-authorship histories and flags suspected dual submissions by comparing manuscripts against submission information from other conferences. Further, the system employs methods to detect low-quality or automated (AI-generated) reviews and pinpoint discrepancies between reviewers' recommendations and final acceptance decisions. ReviewHQ has been used since 2024 across the SIGIR and SIGIR-AP conferences.

CCS Concepts

• **Social and professional topics** → **Professional topics**; • **Information systems** → *Expert search*; *Retrieval models and ranking*.

Keywords

Peer-review systems; Scientific Papers Review Assignment

ACM Reference Format:
Guido Zuccon. 2025. ReviewHQ: An API-Based System for Reviewer Assignment and Quality Control in Research Conferences. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3726302.3730139

1 Introduction

Managing the review process of large-scale academic conferences is a complex task that encompasses multiple stages: collecting submissions, identifying qualified reviewers, assigning reviewers to papers, handling conflicts of interest (COIs), and ensuring the overall quality and fairness of reviews. Conferences have traditionally relied on a combination of manual efforts and semi-automated tools to facilitate these processes. However, as the number of submissions continues to grow, more robust and flexible systems are needed. Matching reviewers to submissions in a way that reflects subject-matter expertise, avoids COIs, and balances reviewer load is a long-standing challenge in the scholarly community [3, 4, 8, 10].

Existing solutions for reviewer assignment typically use either hand-crafted rules or semi-automated approaches to match reviewers based on keyword overlaps, self-declared expertise, or bids (i.e. preferences expressed by reviewers towards reviewing a submission) [5, 12]. While such systems have been instrumental in standardizing the assignment process, they often lack the flexibility to integrate alternative scoring mechanisms and advanced COI detection. Moreover, large conferences must contend with increasingly intricate scenarios, such as submissions involving multi-institutional collaborations, potential dual submissions across different conferences, and the rise of AI-generated reviews [11, 13]. To address these needs, more comprehensive platforms are required.

In this paper, we introduce ReviewHQ, an API-based system designed to streamline and enhance the conference management workflow by providing advanced functionalities beyond basic reviewer-paper matching. While reviewer assignment itself is a well-studied problem [1, 4, 7, 8], ReviewHQ offers additional features to meet the evolving requirements of conferences:

- Configurable Matching Score: The system supports multiple embedding options for representing both reviewers and submissions as dense vectors. Conferences can customize the final matching score by incorporating various similarity measures, reviewer characteristics, and other constraints as needed.
- Advanced Conflict of Interest Mining: Rather than relying solely on author-declared COIs, ReviewHQ mines co-authorship data to automatically identify unreported conflicts. This functionality ensures broader coverage and reduces the burden on authors to exhaustively declare conflicts.
- Review Quality Analysis: The platform detects low-effort and potentially AI-generated reviews by analyzing linguistic cues and consistency metrics. It also highlights discrepancies between meta-reviewers' recommendations and final acceptance decisions, offering insights into possible anomalies.
- Accessory Functions for Conference Management: Additional utilities such as the ability to flag potential dual submissions and analyse the fit of submissions to the program committee.

By consolidating these tasks within one extensible, API-based framework, ReviewHQ aims to reduce overhead for Program Chairs and streamline the entire review process. In the following sections,

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730139

we present the system's core functionalities, discuss the methodologies adopted for expert profile creation and review assignment, and demonstrate how the platform integrates conflict detection and review-quality analytics. We conclude by outlining future directions for extending and refining ReviewHQ. ReviewHQ has been used for SIGIR 2024 and 2025, and for SIGIR-AP 2024.

2 API Overview

ReviewHQ exposes a RESTful API that can be used to manage various tasks involved in conference reviewing, from importing reviewer data to assigning reviewers to submissions and analyzing COIs. A comprehensive list of available functions, including parameter details and usage examples, is provided at: http://reviewhq.ielab.io:8080/docs, and a step-by-step tutorial on how to integrate and use these functions in a custom workflow can be found at http://reviewhq.ielab.io:8080/info; this link also includes information to access a demo of the system.

Core Endpoints. The API is organized around core endpoints that cover the main stages of the conference management workflow:

- *Reviewer Profiles:* Create and update reviewer profiles, fetch existing profiles, and retrieve publications and related metadata from external services such as DBLP and OpenAlex.
- *Submissions:* Upload submissions metadata such as title and abstract, retrieve submission records, and manage updates to previously uploaded content.
- *Conflict Management:* Automate the detection of potential COIs by mining co-authorship data, and manually add or remove reported COIs.
- *Review Assignments:* Compute similarity scores between reviewers and submissions using various embedding methods, set how similarity scores should be mixed with other factors in the assignment of submissions to reviewers, generate a proposed optimal assignment, and retrieve finalized assignments.
- *Review Quality and Integrity Checks:* Inspect reviews to detect issues such as AI-generated text, low-effort feedback, or inconsistencies between reviewers assessment and acceptance decisions.

Authentication and Security. All requests require an API key for authorization. The system provides endpoints for managing these keys; it also logs requests histories. ReviewHQ exploits insights drawn from large language models: these are deployed locally to the system to guarantee confidentiality of submissions.

Integration and Extensibility. The API is designed to be easily integrated into existing conference management platforms. Endpoints return JSON responses and CSV files, making it straightforward to parse the data into other applications or programming frameworks. The documentation at the provided links includes code snippets demonstrating typical usage patterns.

3 Review Assignments

In this section we describe the processes currently performed by ReviewHQ to build proposed reviewers assignments.

3.1 Mining Reviewers Profiles

Program chairs upload program committee members information via a CSV file upload using the `upload_my_reviewers` API endpoint. The API expects information in this file to be organised according to set fields, and the upload methods include data validation routines. The data fields are similar to those that can be extracted from the Easychair system, but contain additional fields related to the URL for the DBLP profile of reviewers.

ReviewHQ allows for the DBLP URL to be left empty: in this case the system will attempt to identify the reviewer in DBLP automatically based on the reviewer name and country[1]. This automatic matching process, however, is not error-free. In particular, for common names where there are multiple DBLP profiles, the algorithm could match to the wrong DBLP profile with the same name.

The DBLP URL is used to mine information about each reviewer. Mining of reviewers' DBLP profiles is initiated via the API endpoint `create_reviewers_profiles`. Among the mined information are (i) the list of affiliations recorded against the reviewer profile, and (ii) a list of the publications the reviewer has authored. The API endpoint accepts two filter parameters, `<year_COI_filter>` and `<year_expertise_filter>`. The `<year_expertise_filter>` indicates how far in the past publications are considered to build the reviewer expertise profile; `<year_COI_filter>` indicates how far in the past publications are considered to automatically identify co-authorship-based COIs. The system then mines publications starting from the earliest of the two years. These filters reduce the amount of mining required: only a subset of authored publications is acquired (from the filter date to the most recent). This reduces the cost of the mining process and execution time: both DBLP and the downstream APIs used to obtain further publication information are rate limited.

Third party API services are then used to acquire the text abstracts of these publications. DBLP in fact does not provide the abstracts. ReviewHQ currently implements methods to interact with Ai2's Semantic Scholar API and the OpenAlex API. Due to recent usage limitations introduced in Semantic Scholar, the default mechanism is relying on OpenAlex. Unlike Semantic Scholar, OpenAlex does not directly contain the abstract of publications; however they can be reconstructed losslessly from the provided inverted index. Publications DOIs extracted from the DBLP data are used to identify the relevant publications in OpenAlex/Semantic Scholar; when a DOI does not exist in DBLP or is not found in the third party API, a title search for exact match is performed.

3.2 Representing Reviewers and Submissions

Submission information is uploaded to ReviewHQ via the API endpoint `upload_my_submissions` as a CSV file, based on the submission.csv file downloadable from Easychair. Unlike reviewers data, submission data (as well as assignment data) is encrypted and cannot be accessed by the system developer to avoid possible concerns of intellectual property leakage and conflict of interest.

The API provides endpoints for creating representations, in the form of embeddings (i.e. dense vectors), for both reviewers and submissions. ReviewHQ currently offers the possibility to create representations using ad-hoc trained `bge` (default) and `stella` (still under validation) encoder models, deployed locally in the ReviewHQ server, thus preserving any sensitive submission information. The system also allows to use OpenAI embedding services to obtain

[1] We also explored the option to use institution affiliation information in early versions, but recognised this has several point of failures: affiliations not updated in Easychair or DBLP, acronyms or shorthands for the affiliation used in DBLP, etc.

dense representations using their web API; however this is discouraged as submission information (which might have sensitive IP value) would be leaked to a third party.

For each reviewer, embeddings of their authored publications (from the previous mining process) are stored, organised in a FAISS index on a per-reviewer basis. Submission and reviewer embeddings are transformed to prevent text reconstruction [9, 15], and transformation keys are encrypted.

3.3 Computing Assignments

Reviewers assignment is modelled as a constrained optimisation problem, for which a solution that maximises the global fit of submissions to reviewer expertise is to be computed. The problem takes as input a matrix $|R| \times |S|$ (|R|: number of reviewers; |S|: number of submissions), and the constrains `<reviewers_per_paper>`, `<min_papers_per_reviewer>`, and `<max_papers_per_reviewer>`. ReviewHQ allows to specify different constrain values across different PC members roles (e.g. different settings across reviewers, senior reviewers, area chairs).

Values in the $|R| \times |S|$ matrix are computed with respect to the similarities between reviewers and submissions representations, and accounting for COIs and other features. The computation of the similarity between a submission and a reviewer is treated as a dense retrieval problem: a reviewer's FAISS index is queried with the submission's dense embedding to retrieve the top k reviewer's publications (k is a configurable parameter). This identifies the k publications the reviewer has authored that are most similar to the submission at hand. The similarity scores of these submissions are combined (options are available to define weights on the linear combination) into a fit score. The fit score is then further modified with a penalty if the affiliation of the reviewer and of any of the submission's author is in the same country. Options are also available for modifying the score to account for whether the reviewer has bid for the paper (e.g., providing a boost). Weights for each of these components are configurable. To eliminate from the optimisation problem reviewers that have a COI with the submission, a large negative number is assigned to the reviewer-submission entry, making this assignment solution impossible to be selected.

Once the matrix has been formed and the bounds of the optimisation problem computed (to check the problem is solvable), the problem is solved using iterative algorithms such as *GLPK_MI*. This produces assignments that are then encrypted and stored in the system. PC Chairs can download decrypted assignments as a CSV using the `download_allocations` API endpoint.

4 Other Functionalities

4.1 Conflict of Interest Mining

Once a list of publications for a reviewer is acquired from DBLP, the bibitems in the list are parsed by ReviewHQ. Along with using this information to acquire publication data for expertise modelling, ReviewHQ uses this also to populate a list of co-authors for automatic COI mining. Automatically mined COIs are complemented with COIs declarations from reviewers/authors downloadable from systems like Easychair. Through the use of ReviewHQ over two years of SIGIR conferences, we have recognised that COIs are often not captured within submissions and review systems. For example, typically SIGIR relies on Easychair for review management, and asks authors and reviewers to self-declare COIs. This process fails at time: Reviewers/authors may fail to identify existing COIs because they do not find conflicted persons within the (often long) list of PC members; or they do not recall having conflict with a specific person at the time of COI declaration. Adding to this is that often COIs need to be entered separately across tracks and conference years, making the COI declaration process frustrating and error prone. The automatic COI discovery capabilities of ReviewHQ provide a (partial) remedy to this, by discovering COIs based on co-authorship. For the Full Track at SIGIR 2025, ReviewHQ identified over 18,000 potential COIs[2] based on co-authorship in addition to those self-declared in Easychairs by authors and reviewers.

The system also allows to identify COIs based on institutions: people at the same institution would be classified as a COI. However, currently this mechanism is brittle. Institution information is manually entered in review management systems, and the same institution could be indicated using different names (e.g. shorthands) by different people; also people might not update their institution in a timely manner when changing job. We have performed initial experimentation with the use of Large Language Models to normalise institution names to a common reference form; however this functionality has not yet been integrated into ReviewHQ.

4.2 Review Quality

ReviewHQ makes available methods for review quality control that PC Chairs can use to improve the review process. Functionalities currently exposed by the API for this are:

- *Flagging AI generated reviews.* For this we have implemented a distributional method and a lexical method [6, 14]. Informal evaluation we performed highlighted the brittleness of the distributional method, that highly depends on the grounding data. We found the lexical method to be more robust: this is based on specifically identified lexical characteristics expressed by popular LLMs when generating text, especially in scientific tasks.
- *Flagging low quality reviews.* For this we have implemented prompts for LLM instances we deployed locally that attempt to identify reviews of low quality, flag these to PC chairs, and provide an explanation of what aspects of the review are identified as being problematic. PC Chairs can upload reviews through the `upload_my_reviews` endpoint. We engineered the prompt with clear instructions of aspects associated with reviews of low quality, distilled from reviewing guidelines, tutorials and practices. Currently ReviewHQ uses LLMs from the Mistral and Llama families for this task.
- *Flagging disagreement between review recommendations and final decisions.* We implemented this process as a quality control step. PC Chairs can upload reviews to ReviewHQ along with final decisions. The system then uses prompts we have engineered to instruct locally deployed LLMs to check the consistency of acceptance decisions with recommendations made by reviewers and meta-reviewers.

4.3 Fit of Submissions to the Reviewing Panel

Large conferences, especially in fields related to Artificial Intelligence, are currently experiencing rapidly increasing number of

[2] Note some of these might be affected by homonyms.

submissions. Through analysing SIGIR 2024 submission data, we identified a significant number of submissions that have no or only peripheral relation to the remit of SIGIR; we believe this could occur with other conferences too. Such out-of-scope submissions are handled differently across different communities, though a common approach is to desk-rejection these submissions to preserve reviewing load for relevant submissions.

To support PC Chairs in identifying submissions that might be out of the scope of the conference, we have implemented retrievability methods [2] based on the dense retrievers employed to compute reviewer-submission fit. An example of such analyses over the submissions and review panel of a recent conference is shown in Figure 1. Armed with this information, PC Chairs could focus their attention to submissions in the tail end of the distribution: these exhibit the least fitness to the expertise of the reviewing panel. This analysis might not only reveal papers that are out-of-scope: it might also identify legit submissions that the current reviewing panel is not equipped to review. In this case, PC Chairs might consider recruiting new reviewers with the relevant expertise.

4.4 Future Avenues of Development

Although the core ReviewHQ API already streamlines many aspects of conference management and reviewer assignments, several enhancements are planned to extend its utility and flexibility.

Web Interface and Reviewer Portal. A dedicated web interface will be developed to provide a user-friendly layer on top of the existing RESTful API. This interface would benefit both PC chairs and reviewers by offering an intuitive dashboard for core tasks.

Self-Management of Reviewer Profiles. An extension of the planned web portal involves enabling reviewers to directly curate their own profiles. By allowing users to remove publications, add newly published work, and declare additional COIs, the system can improve both the completeness and accuracy of reviewer data. This is based on feedback from SIGIR/SIGIR-AP 2024 assignments, where some reviewers were matched with submissions based on a subset of their publications to which they had only minor contributions (e.g., providing specific expertise). As a result, they lacked in-depth familiarity with the broader research topic of these publications. This personalized approach is expected to increase reviewer engagement and reduce the overhead for conference organizers, who currently rely on automated or chair-driven updates to profile information.

Enhanced Encryption with Public-Private Keys. While ReviewHQ already provides an API key–based authentication scheme and encrypts sensible data such as submissions information and review assignments, a public-private key–based mechanism is under development. This mechanism will offer streamlined and more robust security.

Integration with Other Systems. ReviewHQ is designed to be API-first, making it relatively straightforward to integrate with existing conference management systems, e.g., Easychair, ConfTool. Building on this foundation, future efforts will include developing dedicated plug-ins and connectors to simplify data exchange and synchronization with third-party platforms. This integration will reduce duplication of information and enable seamless workflows for submission handling, reviewer allocation, and decision tracking.

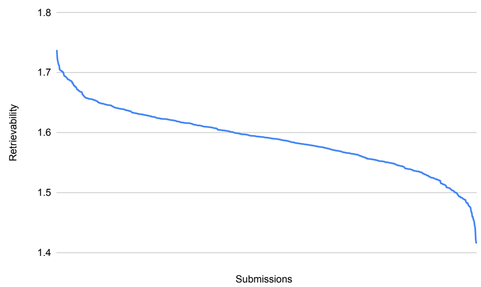

Figure 1: Distribution of retrievability scores across submissions to a recent conference. Low scores indicate low fit to the panel of reviewers.

Advanced Configurable Scoring and Representation Techniques. While the current version of ReviewHQ supports multiple embedding models and permits basic weighting of scores, future developments will offer more sophisticated score configuration options. Organizers will be able to incorporate factors such as paper-related metadata (e.g., domain classification) and historical performance metrics (e.g., review timeliness or quality) into the assignment process. This enhanced configurability will provide conferences with a tailored approach to reviewer matching, maximizing the fairness and expertise alignment of reviewer-paper assignments.

We also plan to increase the representation techniques available in ReviewHQ. Planned enhancements will include devising effective representation learning methods to enhance the quality of open representation models, and integration of LLM-based hybrid representations, such as PromptReps [16], whose sparse mechanism could be exploited to provide a rapid high-level interpretation mechanism for the assignments. Robust evaluation of these techniques in the context of reviewer assignments will also be performed.

These enhancements are expected to make ReviewHQ a more comprehensive, secure, and efficient solution for handling the complexities of academic peer review.

5 Conclusion

This paper introduced ReviewHQ, an API-based platform designed to streamline reviewer assignment and enhance conference management workflows. By integrating diverse capabilities – embedding-based reviewer-paper matching, automated conflict-of-interest detection, and review-quality monitoring – ReviewHQ addresses many of the emerging challenges in peer review processes. Its flexible architecture allows organizers to tailor the assignment process to specific conference needs, while the system's analytics features reduce administrative overhead and ensure greater transparency in the review cycle.

ReviewHQ has already seen practical adoption in SIGIR and SIGIR-AP. Ongoing and future developments, including a user-friendly web portal and deeper integrations with established systems, are intended to further reduce manual intervention and simplify the user experience.

Acknowledgement. The development of ReviewHQ would have not been possible without the trust and feedback from SIGIR 2024 and 2025 and SIGIR-AP 2024 PC Chairs to which we are grateful. We are also grateful to Dr. Teerapong Leelanupab (UQ) for managing the cloud and GPU infrastructure on which ReviewHQ is deployed.

References

[1] Negar Arabzadeh, Sajad Ebrahimi, Sara Salamat, Mahdi Bashari, and Ebrahim Bagheri. 2024. Reviewerly: Modeling the Reviewer Assignment Task as an Information Retrieval Problem. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 5554–5555.

[2] Leif Azzopardi and Vishwa Vinay. 2008. Retrievability: An evaluation measure for higher order information access tasks. In *Proceedings of the 17th ACM conference on Information and knowledge management*. 561–570.

[3] Laurent Charlin, Richard S Zemel, and Craig Boutilier. 2011. A Framework for Optimizing Paper Matching.. In *UAI*, Vol. 11. 86–95.

[4] Susan T Dumais and Jakob Nielsen. 1992. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*. 233–244.

[5] Maryam Karimzadehgan and ChengXiang Zhai. 2009. Constrained multi-aspect expertise matching for committee review assignment. In *Proceedings of the 18th ACM conference on Information and knowledge management*. 1697–1700.

[6] Sandeep Kumar, Mohit Sahu, Vardhan Gacche, Tirthankar Ghosal, and Asif Ekbal. 2024. 'Quis custodiet ipsos custodes?'Who will watch the watchmen? On Detecting AI-generated peer-reviews. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 22663–22679.

[7] Kevin Leyton-Brown, Yatin Nandwani, Hedayat Zarkoob, Chris Cameron, Neil Newman, Dinesh Raghu, et al. 2024. Matching papers and reviewers at large conferences. *Artificial Intelligence* 331 (2024), 104119.

[8] David Mimno and Andrew McCallum. 2007. Expertise modeling for matching papers with reviewers. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. 500–509.

[9] John Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M Rush. 2023. Text Embeddings Reveal (Almost) As Much As Text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 12448–12460.

[10] Lorena Parra, Sandra Sendra, Sara Ficarelli, and Jaime Lloret. 2013. Comparison of Online Platforms for the Review Process of Conference Paper. In *CONTENT 2013, The Fifth International Conference on Creative Content Technologies*. IARIA XPS Press, 16–22.

[11] David Soergel, Adam Saunders, and Andrew McCallum. 2013. Open Scholarship and Peer Review: a Time for Experimentation. (2013).

[12] David Tran, Alex Valtchanov, Keshav Ganapathy, Raymond Feng, Eric Slud, Micah Goldblum, and Tom Goldstein. 2020. An open review of openreview: A critical analysis of the machine learning conference review process. *arXiv preprint arXiv:2010.05137* (2020).

[13] Gang Wang, Qi Peng, Yanfeng Zhang, and Mingyang Zhang. 2023. What have we learned from OpenReview? *World Wide Web* 26, 2 (2023), 683–708.

[14] Xianjun Yang, Liangming Pan, Xuandong Zhao, Haifeng Chen, Linda Petzold, William Yang Wang, and Wei Cheng. 2023. A survey on detection of llms-generated content. *arXiv preprint arXiv:2310.15654* (2023).

[15] Shengyao Zhuang, Bevan Koopman, Xiaoran Chu, and Guido Zuccon. 2024. Understanding and mitigating the threat of vec2text to dense retrieval systems. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. 259–268.

[16] Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *arXiv preprint arXiv:2404.18424* (2024).