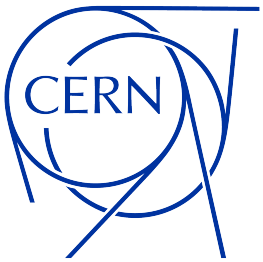
NeutrinoReview 2.0:

UI Redesign, System Stabilization, and Scientific Evaluation

Author: Uday Sharma [1,2]

A report presented as part of the

CERN Summer Student Programme 2025



Supervisors

Andreas Wagner[2]
Elias Sandner[2,3]

European Organization for Nuclear Research (CERN)
IT Department

5th September 2025

[1] University of Delhi
[2] European Organization for Nuclear Research (CERN)
[3] Graz University of Technology (TU Graz)

Abstract

This report details the development and research work undertaken on the NeutrinoReview application, an AI-powered tool to automate systematic reviews to conduct evidence based research during the 2025 CERN Summer Student Programme. The project began with a comprehensive overhaul of the system, including a complete re-architecture and redesign of all web components, significantly improving the application's design and simplifying the user workflow. To modernize the development lifecycle, the application was containerized using Docker, a full CI/CD pipeline was established in GitLab, and the first version was successfully deployed on CERN OKD infrastructure, integrated with CERN Single Sign-On, and validated through user studies. Key algorithmic improvements were made, including debugging and enhancing the data deduplication module, creating a new data ingestion pipeline from ArXiv, and implementing a novel LLM Screening method called Cal-X. Additionally, research was conducted to validate the system's performance, with findings published in a research paper. Additional studies evaluated the reasoning capabilities of various Large Language Models laying the groundwork for future publication. This work has transformed NeutrinoReview into a robust, user-friendly, and scalable platform for scientific literature analysis.

Keywords: Artificial Intelligence, Systematic Review, Large Language Models (LLMs), Machine Learning, Software Development, Summer Student

Contents

List of Figures

List of Tables

1 Introduction

Systematic reviews are the cornerstone of evidence-based research, providing exhaustive and impartial summaries of existing studies on specific topics and are considered the most reliable form of scientific evidence. These comprehensive analyses help researchers and practitioners make informed decisions by synthesizing all available evidence on a particular research question.

However, the exponential growth of scientific literature poses significant challenges for researchers conducting systematic reviews. The time-intensive process of extracting and managing data from hundreds or thousands of papers can lead to researcher exhaustion and potential compromises in review quality. This is particularly problematic in medical research, where the volume of publications continues to increase at an unprecedented rate, making it increasingly difficult for researchers to stay current with the latest findings.

1.1 The NeutrinoReview Project

CERN[1] and the World Health Organization (WHO)[2] collaborate on a joint venture called the CAiMIRA/ARIA[3] project, with work package 4 addressing these challenges. Through this partnership, a conceptual design for NeutrinoReview was created, a systematic review AI tool designed to simplify the way systematic reviews are conducted [3].

NeutrinoReview streamlines the systematic review process through an automated workflow pipeline. The tool begins by fetching research data from various databases using user-defined search strings, followed by automated deduplication to remove redundant studies. It then employs artificial intelligence to perform initial screening based on predefined eligibility criteria, before passing the results to researchers for final manual validation. This complete workflow significantly reduces the manual workload while maintaining the rigor required for high-quality systematic reviews.

The primary target users are medical researchers who face the growing challenge of managing an ever-increasing volume of scientific publications in their systematic review work. However, the underlying technology is not restricted to medical research and also suitable for evidence synthesis in any other field.

1.2 Summer Project

At the beginning of the CERN Summer Student Programme, the NeutrinoReview project presented significant opportunities for improvement. The system was limited to local development environments with no deployed version available, restricted to medical research databases, and suffered from various technical limitations including non-functional CERN Single Sign-On (SSO), lack of containerization, and a static PRISMA diagram visualization system. The user interface required substantial streamlining, and there was no continuous integration and deployment (CI/CD) pipeline in place.

Due to resource constraints and isolated ownership over time, several critical tasks had accumulated, creating the perfect opportunity for a comprehensive summer project. My role throughout the summer was to transform NeutrinoReview from its existing state into the first proper working prototype—a system that could be showcased and developed into a production-ready tool. The main objectives of my summer project were organized into five key areas: Codebase Refactoring and System Modernization, Redesign of the Entire Application, Automating the Development-to-Deployment Process, Enhancing NeutrinoReview with New Capabilities, and Research Contributions and Academic Writing.

This report documents the comprehensive transformation of NeutrinoReview during my summer at CERN. The following sections detail the system modernization and architectural improvements (Section 2), the algorithmic enhancements and data processing developments (Section 3), the experimental evaluation of these improvements (Section 4), a discussion of the results and their broader implications (Section 5), and concluding remarks with suggestions for future development (Section 6).

[1] https://home.cern/
[2] https://who.int/
[3] https://partnersplatform.who.int/tools/aria/

Through this work, NeutrinoReview has evolved into a robust, user-friendly platform that shows significant promise. A user study demonstrated that the data retrieval and deduplication phase can be conducted 4 times faster than with traditional tools while maintaining quality and intuitive usability [1]. Furthermore, the implemented screening algorithms have demonstrated significant workload reduction while maintaining high sensitivity [2].

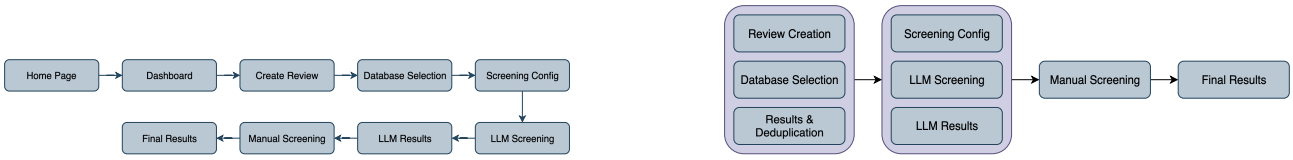
2 System Modernization and Architectural Overhaul

2.1 Full-Stack Re-architecture and UI/UX Redesign

At the start of the summer, NeutrinoReview faced significant architectural and usability challenges. While the core technology stack—consisting of React for the frontend, Python with FastAPI for the backend, and PostgreSQL with SQLAlchemy for data management—provided a solid foundation, the implementation suffered from numerous critical issues.

The application had bugs due to resource constraints and isolated ownership over time. You can find this documented extensively in the GitLab issue board[4], including problems with custom file upload functionality, infinite requests, widespread UI inconsistencies and many more. More fundamentally, the user workflow was fragmented and non-intuitive, requiring users to navigate through up to 9 different pages to complete a single systematic review. Critical functions were scattered across multiple interfaces: review management on one page, database selection on another, and eligibility criteria configuration on yet another page. This fragmented approach created unnecessary complexity and significantly hindered the user experience.

Rather than completely rebuilding the technology stack, focus was on restructuring the application architecture to create a more cohesive and intuitive user experience. The core improvement involved consolidating the fragmented workflow into a streamlined 4-page structure as shown in Fig. 1.



(a) Old Application Flow (b) New Application Flow

Figure 1: Comparison of the Old and New Application Flow

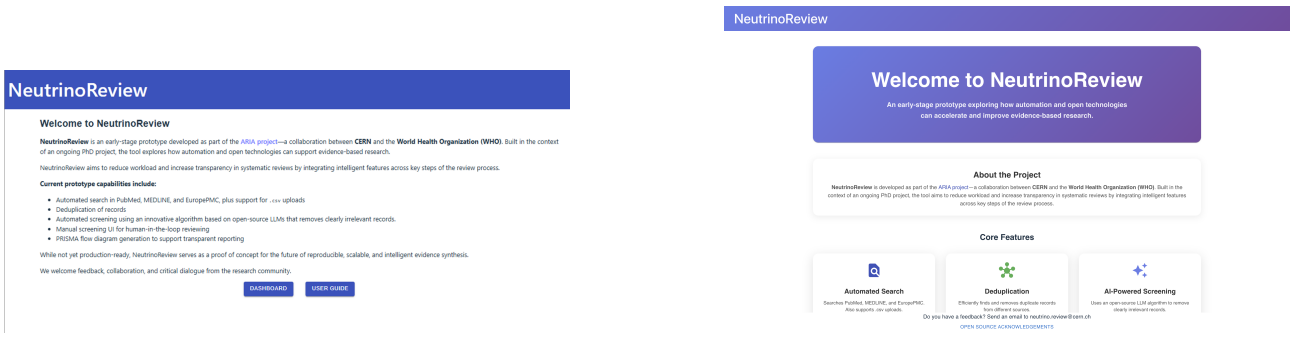
This architectural consolidation eliminated the need for users to constantly navigate between disconnected pages while maintaining all the essential functionality of the original system. All the pages were redesigned from scratch including all the different components to comply with the design scheme. All React component in the client appear in the diff from when the summer studentship began to HEAD, confirming that every component in addition to related CSS, JS, and asset files was modified and rebuilt. The detailed explanation can be found in the Section A. This part of the project focused on first fixing all the bugs and then fixing the UI/UX by redesigning all the components and pages to make it more simple and intuititve. Screenshots of the new and old application can be found at the end of this report in Fig. 7 and Fig. 6 respectively.

2.2 Implementation of a Modern DevOps Workflow

2.2.1 Containerization with Docker

Prior to containerization, the development setup for NeutrinoReview required developers to manually configure and run multiple services: starting a React development server, launching a FastAPI backend, and setting up pgAdmin locally. This approach created significant challenges as developers had to ensure version compatibility across all components and maintain consistent development environments.

[4] https://gitlab.cern.ch/caimira/caimira-wp4/neutrinoreview



(a) New Landing Page (b) New Landing Page

Figure 2: New and Old Application Landing Pages

More critically, there was no reliable production deployment system. While a manual deployment had been attempted previously, the production environment had broken, and there was no systematic way to deploy new versions. This lack of infrastructure meant that NeutrinoReview existed only as a local development application with no accessible live version for test users or stakeholders.

To address these challenges, a comprehensive containerization strategy using Docker was implemented. Dedicated Dockerfiles for both the client and server components, enabling the entire development environment to be spun up with a single command was created. Additionally, production-ready container images optimized for deployment environments were developed. This containerization eliminated environment inconsistencies and significantly simplified both development setup and production deployment processes.

2.2.2 Continuous Integration and Deployment (CI/CD) with GitLab

Building on the containerized infrastructure, a complete CI/CD pipeline using GitLab to automate the development-to-deployment workflow was established. The pipeline was designed to eliminate manual deployment processes and ensure consistent, reliable releases.

The CI/CD pipeline consists of several automated stages. First, separate build jobs use Buildah to create optimized container images for both the server and client components. These images are automatically pushed to the GitLab container registry upon successful builds. The pipeline then triggers an automatic redeployment on the CERN OKD infrastructure whenever new commits are pushed to the master branch, ensuring that the latest version is always available to users.

This automation transformed the development workflow from manual, error-prone deployments to a streamlined process where code changes are automatically tested, built, and deployed. The pipeline significantly reduced deployment time and eliminated the risk of human error in the release process.

2.2.3 Deployment on the CERN OKD Infrastructure

CERN OKD (OpenShift Kubernetes Distribution) is CERN's container orchestration platform based on Red Hat OpenShift, designed for deploying and managing containerized applications within the CERN infrastructure. Deploying NeutrinoReview on this platform required learning OpenShift concepts, container orchestration, and CERN-specific deployment procedures.

Two primary services on CERN OKD: the NeutrinoReview client and server applications were successfully deployed. For database requirements, CERN's Database on Demand (DBoD) service to host the PostgreSQL database was integrated, ensuring reliable data persistence and backup capabilities. The deployment was configured to work seamlessly with the CI/CD pipeline, automatically pulling and deploying new container images as they become available.

One of the significant challenges was configuring external access to allow users outside the CERN network to access NeutrinoReview, which was essential given the collaboration with WHO. This required understanding CERN's network security policies and implementing appropriate access controls.

The successful deployment on CERN OKD represented a major milestone—it was the first live, accessible version of NeutrinoReview available to users and stakeholders. This deployment provided

a stable, scalable platform that could handle multiple concurrent users and serve as a foundation for further development and user testing.

2.3 Integration with CERN Authentication Services (SSO)

NeutrinoReview's integration with the CERN Single Sign-On (SSO) was non-functional. Users attempting to authenticate were not properly redirected back to the application after completing the login process, effectively preventing anyone from accessing the system through CERN's authentication infrastructure.

Resolving this issue required a deep dive into CERN's SSO documentation and authentication protocols. I learned the OAuth 2.0 flow implementation, callback URL configuration, and the specific requirements for integrating with CERN's identity management system in the process. Through systematic debugging and testing, redirect configuration issues that were preventing successful authentication were identified and resolved.

A critical aspect of the SSO implementation involved configuring appropriate access permissions. Since NeutrinoReview is a collaborative tool intended for use by WHO researchers and other external partners, we created a new CERN group called "neutrinoreview-users" that enables access for users outside the CERN network. This required understanding CERN's group management system and coordinating with CERN IT services to establish the necessary permissions.

The successful CERN SSO integration provides several key benefits. It offers robust security out of the box without requiring the NeutrinoReview application to maintain its own user authentication system. Users can seamlessly access the application using their existing CERN credentials, eliminating the need for separate account creation and password management. For the development team, this integration reduces security maintenance overhead and ensures compliance with CERN's security standards.

The working SSO system was essential for enabling user testing and stakeholder demonstrations, as it provided the secure access control necessary for a production-ready application within the CERN ecosystem.

3 Algorithmic Enhancements and Data Processing

Following the successful system modernization and deployment, the next phase focused on expanding NeutrinoReview's capabilities through algorithmic improvements and new data processing features. These enhancements were designed to broaden the tool's applicability beyond medical research and improve the accuracy and efficiency of the systematic review process.

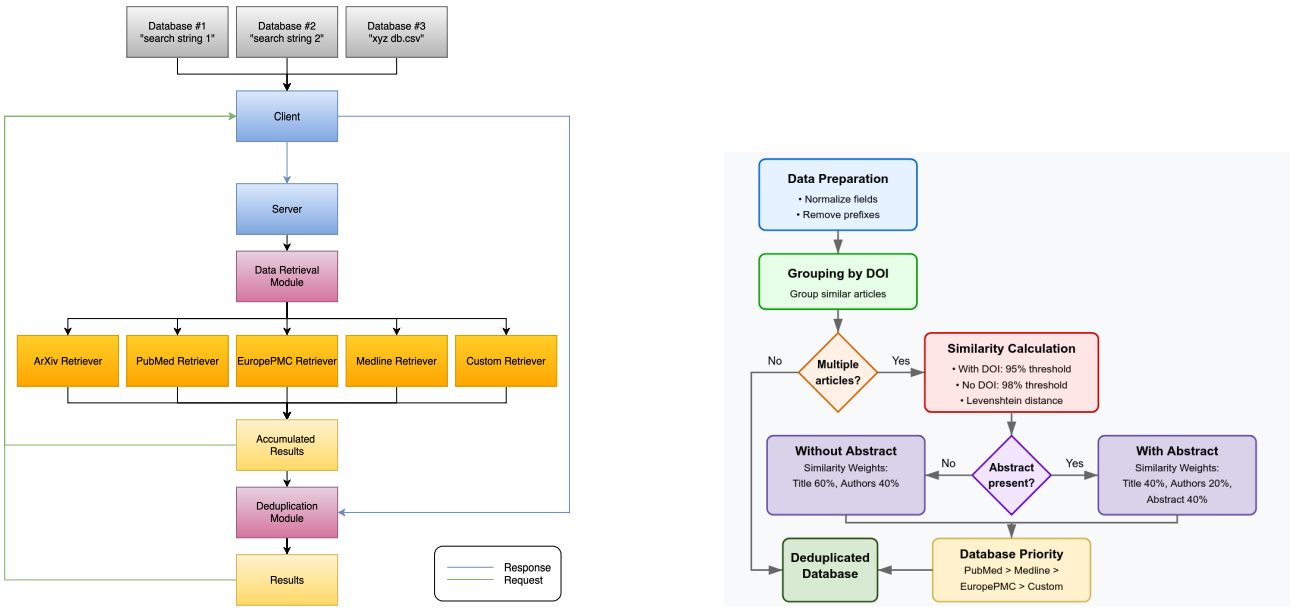
3.1 Automated Data Ingestion from ArXiv

Prior to this enhancement, NeutrinoReview was limited to medical research databases, significantly restricting its applicability to other scientific domains. While users could upload custom CSV files, there was no direct integration with ArXiv, one of the most widely used preprint databases across multiple scientific disciplines.

ArXiv presents unique challenges for automated data retrieval as it does not provide a traditional export function for search results. Instead, researchers must rely on programmatic access through APIs, which requires handling XML data formats, managing API rate limits, and dealing with potential service interruptions.

A comprehensive data pipeline to integrate ArXiv into NeutrinoReview's Data Retrieval and Deduplication (DR&D) module was designed and implemented as shown in Fig. 3a. This pipeline handles all aspects of ArXiv interaction, including API communication, XML processing, error handling, and data normalization to ensure compatibility with the existing system architecture. The integration maintains the same user experience as the existing medical database connections while handling the technical complexities of ArXiv access behind the scenes.

This expansion significantly broadened NeutrinoReview's potential user base, enabling systematic reviews in physics, computer science, mathematics, and other fields represented in ArXiv. The imple-



(a) DR&D Module Design (b) Deduplication Algorithm Design

Figure 3: Data Retrieval and Deduplication (DR&D) Module

mentation demonstrates the system's flexibility and establishes a framework for integrating additional specialized databases in the future.

3.2 Enhancement of the Data Deduplication Module

The original deduplication algorithm, while functional, contained several logical flaws that limited its effectiveness. The existing approach followed a hierarchical database priority system (PubMed > Medline > EuropePMC > Custom) and used a two-stage process illustrated in Image 2: initial grouping by Digital Object Identifier (DOI) followed by text similarity analysis using Levenshtein distance.

However, the implementation suffered from three critical issues that reduced its accuracy:

- Incomplete comparison logic: After DOI-based grouping, papers without DOIs were only compared against papers with unique DOIs, missing potential duplicates within the non-DOI group itself.
- Missing cross-comparison: Papers lacking DOIs were not comprehensively compared against each other, allowing duplicates to persist in the final dataset.
- Flawed similarity matching: A logical error in the similarity matching function incorrectly added higher-priority references to the duplicates list instead of preserving them.

Redesign of the deduplication algorithm was done to address these fundamental issues. The corrected approach implements a more systematic process: first handling DOI-based deduplication while preserving the highest-priority paper from each DOI group, then creating a comprehensive candidate set of all surviving papers (both unique DOI papers and non-DOI papers), and finally running similarity matching across this complete candidate set to ensure every paper is properly evaluated against every relevant comparison.

The enhanced algorithm as shown in Fig. 3b improved deduplication performance significantly. During user studies, the corrected deduplication process identified approximately 50% more duplicate records in systematic reviews compared to the original implementation—a substantial improvement that directly impacts the quality and efficiency of the review process. This enhancement reduces the manual workload for researchers while ensuring more accurate identification of duplicate studies, leading to higher-quality systematic reviews.

3.3 Cal-X: A Novel LLM-based Screening Method

The original NeutrinoReview implementation included only one LLM screening technique: the 5-tier approach, which assigned papers a relevancy score from 1 to 5, with 1 representing the highest relevance [4]. While effective, this approach provided limited flexibility and did not fully leverage the probabilistic outputs available from modern language models.

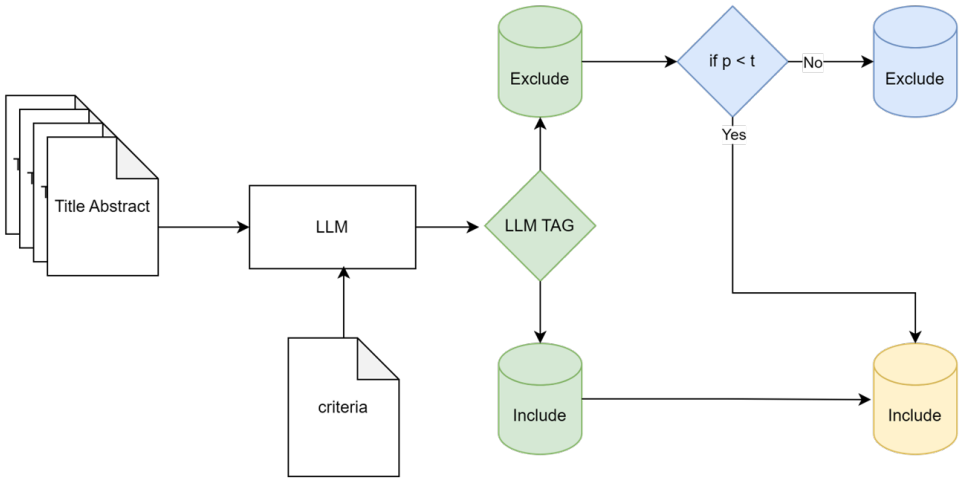


Figure 4: Cal-X Algorithm Design [2]

To address these limitations, we implemented Cal-X, a novel LLM-based screening method that introduces a more sophisticated approach to automated paper screening. The Cal-X method [2] operates through a two-stage process illustrated in Fig. 4. First, it performs binary classification using an LLM to categorize papers as either "include" or "exclude" based on the provided eligibility criteria. This initial classification provides a clear baseline decision for each paper.

The innovation of Cal-X lies in its second stage, which introduces a user-configurable threshold parameter that leverages the probabilistic nature of LLM outputs. The system extracts log probabilities from the LLM's classification decisions and compares them against the threshold. Papers initially classified as "include" are placed in the included dataset. However, papers classified as "exclude" undergo additional scrutiny, if their exclusion probability falls below the threshold, indicating uncertainty in the exclusion decision, they are moved to the included pile rather than being excluded outright.

This probabilistic approach allows researchers to adjust the sensitivity of the screening process based on their specific requirements. A lower threshold results in more conservative screening (fewer papers excluded), while a higher threshold produces more aggressive filtering. This flexibility is particularly valuable in systematic reviews where the cost of missing relevant papers far exceeds the cost of including potentially irrelevant ones.

The implementation of Cal-X required significant development work across multiple system components. New React components to allow users to configure the threshold parameter were created, implemented the core algorithmic logic in the server application, and added new database migrations to store the additional metadata generated by this approach.

4 Experimental Evaluation and Results

To validate the improvements made to NeutrinoReview and assess its performance against established tools, comprehensive evaluations were conducted across multiple dimensions. These evaluations focused on both the technical performance of individual components and the overall user experience of the integrated system.

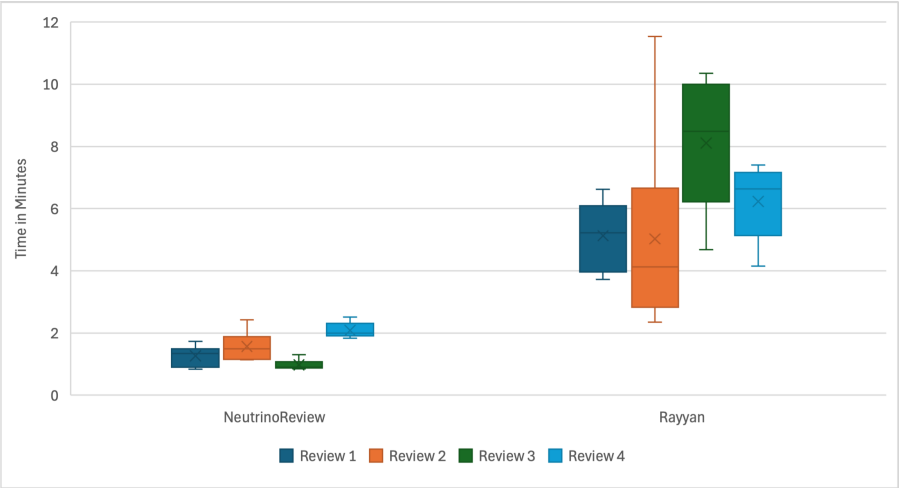


Figure 5: Average Time NeutrinoReview vs Rayyan

4.1 Evaluation of the Data Retrieval and Deduplication (DR&D) Module

To evaluate the time savings achieved through the implemented DR&D module, a user study was conducted. The module's implementation and the study findings reported in this section are also described in a scientific paper currently under review for the 7th International Open Search Symposium[5] [1].

4.1.1 Study Design & Methodology

To evaluate NeutrinoReview's efficiency and usability, a comprehensive user study was conducted comparing it against Rayyan, a widely used conventional systematic review tool. The study involved seven participants, all novices with no prior experience using either system, to ensure unbiased results and eliminate the influence of tool-specific expertise.

Each participant performed the complete data retrieval and deduplication workflow for four distinct medical systematic reviews using both tools. To ensure standardized comparison conditions, participants were provided with predefined search strings for PubMed, Medline, and EuropePMC databases, taken directly from original published reviews to closely mimic real-world systematic review scenarios.

The experimental protocol was carefully designed to maintain consistency across all participants. Initially, participants received written instructions detailing the required steps and attended demonstrations of the systematic review workflow in both NeutrinoReview and Rayyan to establish a consistent baseline understanding. During the task-performance phase, a facilitator was present to observe and provide clarification upon request, preventing procedural impasses. This "assisted completion" protocol ensured that recorded task times primarily reflected each tool's inherent efficiency rather than user confusion or errors.

For valid comparison, Rayyan's deduplication process was configured to emulate NeutrinoReview's automated algorithm using Rayyan's "AutoResolver" feature. A multi-step logic was established that mirrored NeutrinoReview's process, including replication of the database priority hierarchy (PubMed $>$ Medline $>$ EuropePMC) and conflict resolution rules that sequentially checked for duplicates by DOI, normalized title and author, and applied a 95% similarity threshold.

4.1.2 Results and Performance Analysis

The results demonstrated substantial performance advantages for NeutrinoReview across all evaluation metrics. All participants completed tasks more quickly with NeutrinoReview than with Rayyan, with remarkably consistent results. Notably, even the slowest participant using NeutrinoReview outperformed the fastest participant using Rayyan in 3 out of 4 test reviews.

Quantitatively, task completion with NeutrinoReview required approximately 1.5 minutes on average, while the same tasks took more than 6 minutes with Rayyan. This represents a 4-fold increase

[5] https://opensearchfoundation.org/en/events-osf/ossym25/

in speed and an overall time reduction of 75%. These findings indicate that the automated workflow implemented in NeutrinoReview's DR&D module substantially reduces the effort required for data retrieval and deduplication in systematic reviews.

4.2 Extended Evaluation of 5-Tier Approach

4.2.1 Model Selection and Evaluation Framework

Following the acceptance of the initial work at IEEE FLLM 2024 [4] and subsequent invitation to extend the paper for the Information Processing & Management Journal, comprehensive evaluations of the 5-tier LLM screening approach were conducted using multiple state-of-the-art language models.

The evaluation began with a systematic analysis of available LLMs to determine the most suitable options within budget constraints. As detailed in Table 1 of the model evaluation report, several models were assessed based on context window size, completion capabilities, and usability for systematic review tasks. Models with insufficient context windows (such as the Llama prompt guard variants with only 512 tokens) were eliminated as unsuitable for processing full academic abstracts and eligibility criteria.

Table 1: Analysis of Available AI Models and Their Usability

Model ID	Developer	Context Window	Max Completion
deepseek-r1-distill-llama-70b	DeepSeek / Meta	131,072	131,072
meta-llama/llama-4-maverick-17b-128e-instruct	Meta	131,072	8,192
meta-llama/llama-4-scout-17b-16e-instruct	Meta	131,072	8,192
meta-llama/llama-prompt-guard-2-22m	Meta	512	512
meta-llama/llama-prompt-guard-2-86m	Meta	512	512
moonshotai/kimi-k2-instruct	Moonshot AI	131,072	16,384
openai/gpt-oss-120b	OpenAI	131,072	65,536
openai/gpt-oss-20b	OpenAI	131,072	65,536
playai-tts	PlayAI	8,192	8,192
playai-tts-arabic	PlayAI	8,192	8,192
qwen/qwen3-32b	Alibaba Cloud	131,072	40,960

Performance benchmarking was conducted using two established evaluation frameworks. LLM Arena rankings, based on crowdsourced anonymous pairwise comparisons, indicated that `gpt-oss-120b` and `deepseek-r1-distill-llama-70b` performed best on instruction-following tasks, with overall rankings of 14 and 36 respectively. LiveBench [5] reasoning evaluations revealed that `qwen3-32b` (83.08 reasoning average) and `gpt-oss-120b` (77.61 reasoning average) demonstrated superior reasoning capabilities.

The experiments were carried out on 3 models. Based on the above analysis and considering the consistent performance of GPT-based models across both benchmarking platforms `gpt-oss-120b` was selected. The other model we chose for our study was `meta-llama/llama-4-maverick-17b-128e-instruct` because, although only 17B parameters are active during inference, the model is part of a larger mixture-of-experts architecture with 400B total parameters. We also used `llama-3.5-8b` to continue a previous study [4].

4.2.2 Experimental Design and Implementation

The experimental framework evaluated each model's performance on the 5-tier screening task, assessing accuracy, consistency, and reliability across diverse systematic review domains. This extensive evaluation provides robust evidence for the effectiveness of LLM-based screening approaches and establishes performance baselines for different model architectures and sizes.

The evaluation comprised of 30 systematic review datasets for `llama-4-maverick-17b-128e-instruct` and `gpt-oss-120b`. The experiment also consisted testing of 13 datasets using locally deployed `models/llama-3.5-8b` to extend the study done previously with only 17 of the 30 systematic reviews.

This comprehensive evaluation required implementing asynchronous processing systems to handle the large volume of API requests efficiently while maintaining data integrity and experimental consistency.

The extended evaluation required to overcome significant technical challenges to manage computational costs while maintaining experimental rigor. The experiments were redesigned to utilize Groq Cloud's batch processing capabilities rather than standard API requests, substantially reducing costs while enabling large-scale evaluation.

Due to the termination of the internship, the analysis of the experimental results was left to colleagues in the project team. The results of this extended evaluation are expected to contribute significantly to understanding the capabilities and limitations of the 5-tier approach, providing valuable insights for future development and deployment of AI-powered research tools.

5 Discussion

5.1 Impact of System Modernization

The comprehensive modernization of NeutrinoReview has fundamentally transformed it from a fragmented, locally-bound application into a robust, user-friendly, and scalable platform suitable for scientific research. The architectural overhaul successfully addressed the critical limitations that previously hindered the tool's usability and accessibility.

The consolidation of the user workflow from 9 disparate pages to a streamlined 4-page structure represents more than a superficial interface change—it reflects a fundamental shift toward user-centered design that aligns with how researchers naturally conduct systematic reviews. The redesign of all the web components not only resolved numerous bugs documented in the GitLab issue boards but also created a cohesive visual and functional experience that significantly reduces the cognitive load on users.

The implementation of modern DevOps practices has established a foundation for sustainable development and deployment. The containerization with Docker eliminated environment inconsistencies that previously plagued development, while the CI/CD pipeline transformed deployment from a manual, error-prone process to an automated, reliable workflow. The successful deployment on CERN's OKD infrastructure represents a crucial milestone—the first live, accessible version of NeutrinoReview available to users and stakeholders.

The integration of CERN Single Sign-On authentication not only resolved technical functionality issues but also positioned NeutrinoReview within CERN's secure infrastructure ecosystem. The creation of the "NeutrinoReview-users" group enables seamless collaboration with external partners like WHO while maintaining appropriate security controls.

5.2 Significance of Algorithmic Improvements

The algorithmic enhancements implemented during this project address fundamental limitations in systematic review automation and demonstrate significant practical impact. The enhanced deduplication module's identification of approximately 50% more duplicate records compared to the original implementation represents a substantial improvement in data quality. This enhancement directly reduces researcher workload by ensuring more accurate datasets for subsequent screening phases, ultimately improving the reliability of systematic review outcomes.

The implementation of the Cal-X screening method introduces a novel approach to LLM-based paper screening that leverages probabilistic outputs to provide researchers with greater control over screening sensitivity. Unlike the existing 5-tier approach, Cal-X allows users to adjust the confidence threshold based on their specific research requirements, enabling more nuanced screening decisions. This flexibility is particularly valuable in systematic reviews where the cost of missing relevant papers significantly exceeds the cost of including potentially irrelevant ones.

The integration of ArXiv as a data source represents a strategic expansion of NeutrinoReview's applicability beyond medical research. This enhancement broadens the potential user base to include researchers in physics, computer science, mathematics, and other fields, demonstrating the system's adaptability and establishing a framework for future database integrations.

5.3 Broader Implications

The user study results revealing a 4-fold improvement in task completion time compared to Rayyan highlight the potential impact of automation-focused design in systematic review tools. While these results must be interpreted within the context of comparing a proof-of-concept tool with an established platform, they demonstrate the significant efficiency gains possible through integrated, automated workflows.

The extended evaluation of multiple LLM models for systematic review screening provides valuable insights into the current capabilities and limitations of AI-powered research tools. The systematic analysis of model performance across different benchmarks (LLM Arena and LiveBench) establishes a methodology for selecting appropriate models for specific research applications and contributes to the broader understanding of LLM effectiveness in scientific literature analysis.

The successful transformation of NeutrinoReview from concept to functional prototype demonstrates the feasibility of developing sophisticated research tools within academic environments. The project illustrates how modern software development practices can be effectively applied to scientific tool development, creating sustainable, maintainable systems that can evolve with changing research needs.

6 Conclusion & Future Work

6.1 Conclusion

This CERN Summer Student Programme project successfully transformed NeutrinoReview from a bug-ridden, locally-constrained application into a functional, deployed prototype that demonstrates significant potential for revolutionizing systematic review processes. The comprehensive modernization encompassed system architecture, user interface design, deployment infrastructure, and algorithmic capabilities.

The key achievements of this project include the complete re-architecture and redesign of all of the web components, creating an intuitive 4-page workflow that consolidates previously fragmented functionality. The implementation of modern DevOps practices, including Docker containerization, GitLab CI/CD pipelines, and deployment on CERN OKD infrastructure, established the first live, accessible version of NeutrinoReview. The integration of functional CERN SSO authentication enables secure access for both CERN users and external collaborators.

Algorithmic enhancements significantly improved the tool's capabilities and accuracy. The enhanced deduplication algorithm identifies approximately 50% more duplicate records than the previous implementation, while the novel Cal-X screening method provides researchers with greater flexibility in controlling screening sensitivity through probabilistic thresholds. The integration of ArXiv as a data source expanded the tool's applicability beyond medical research to multiple scientific domains.

The experimental evaluations validate these improvements, demonstrating a 4-fold increase in task completion speed compared to conventional tools and establishing performance benchmarks for different LLM models in systematic review screening tasks. These results provide strong evidence for the potential of automation-focused approaches in scientific literature analysis.

The project deliverables include a testing-ready tool with full CI/CD functionality, comprehensive documentation, and a foundation for future development. While not yet production-ready, the current prototype successfully demonstrates all core functionalities and provides an excellent foundation for continued development toward a production system.

6.2 Future Work

Several strategic directions emerge for the continued development of NeutrinoReview toward a production-ready systematic review platform:

- Full Text Retrieval & Screening: Expanding beyond abstract-based screening to incorporate full-text analysis would significantly enhance the tool's screening accuracy and completeness. This

enhancement would require developing robust methods for full-text acquisition, processing, and analysis.

- Production-Scale Deduplication Algorithm Optimization: While the current deduplication algorithm demonstrates improved accuracy, production deployment with large datasets will require optimization for time complexity and computational efficiency. Implementing more sophisticated algorithms and potentially incorporating machine learning approaches.
- Collaboration Features: Systematic reviews typically involve multiple researchers working collaboratively. Implementing features for team management, conflict resolution, and collaborative decision-making would enhance the tool's practical utility for research teams.
- Enhanced System Architecture for Large-Scale Production: The current architecture, while suitable for prototype demonstration, will require substantial enhancement to support production-scale usage. This includes implementing robust load balancing, horizontal scaling capabilities, enhanced security measures, comprehensive monitoring and logging systems, and optimized database architectures for handling large volumes of concurrent users and data.

These future developments will build upon the solid foundation established during this summer project, ultimately realizing the vision of a comprehensive, AI-powered systematic review platform that can significantly accelerate evidence-based research across multiple scientific domains.

7 Acknowledgements

I would like to thank Elias and Andreas for being my supervisors and helping and guiding me throughout my summer studentship with not only problems relating to the project but everyday problems as well. They were the best supervisors I could have asked for who were there to help me at every step of my internship.

I would also like to thank the entire IT-PW-WA section for accommodating me for the summer. It was really fun being part of the section and going to all the barbecues. I am grateful to the CERN IT services team who assisted with the OKD deployment and infrastructure setup. I extend my gratitude to the WHO collaborators in the CAiMIRA/ARIA project and all the participants who volunteered for the user studies, whose contributions were essential to validating the system's performance.

I am also very thankful to all the friends I made at CERN, especially Martin and Harold, who made my first time abroad really fun. The entire summer student community created a supportive and inspiring environment that enhanced both my personal and professional growth. Finally, I would like to thank the CERN Summer Student Programme organizers for this incredible opportunity, and my family for their support in enabling me to participate in this transformative experience.

References

[1] Elias Sandner et al. "Automating Systematic Reviews: API-Powered Bibliographic Data Retrieval Module for NeutrinoReview". Manuscript under review at the 7th International Open Search Symposium (OSSYM 2025). 2025. URL: https://opensearchfoundation.org/en/events-osf/ossym25/.

[2] Elias Sandner et al. "Cal-X: Enhancing Systematic Review Screening with LLMs and Next-Token Likelihood Calibration". Manuscript under review at the 3rd International Conference on Foundation and Large Language Models (FLLM 2025). 2025. URL: https://fllm-conference.org/2025/.

[3] Elias Sandner et al. "NeutrinoReview: Concept Proposal for An Open Source Review Management Tool". In: *6th International Open Search Symposium*. 2024, p. 53.

[4] Elias Sandner et al. “Screening Automation for Systematic Reviews: A 5-Tier Prompting Approach Meeting Cochrane’s Sensitivity Requirement”. In: *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*. 2024, pp. 150–159. DOI: 10.1109/FLLM63129.2024.10852425.

[5] Colin White et al. “LiveBench: A challenging, contamination-limited LLM benchmark”. In: *arXiv preprint arXiv:2406.19314* (2024).

A Code Snippets

```
5_TIER_PROMPT = """Instructions: You are a researcher rigorously screening titles and
abstracts of scientific papers for inclusion or exclusion in a review paper.
Based on the provided inclusion and exclusion criteria listed below, you are
asked to assign the paper to one of the following groups:

"1 Highly Relevant": Based on the given title and abstract, the paper meets all
inclusion criteria and no exclusion criteria. Therefore, the paper will be
included.
"2 Probably Relevant": The information provided in the title and abstract indicates
that the paper is likely relevant.
"3 Undecidable": The given title and abstract do not contain enough information to
evaluate whether the inclusion and exclusion criteria are met.
"4 Probably Irrelevant": Based on the given title and abstract, it is likely that at
least one inclusion criterion is not met or that at least one of the exclusion
criteria is met.
"5 Not Relevant": Based on the title and abstract, it is clear that the paper does
not meet the criteria. Therefore, the paper will be excluded.

Based on the probability of a paper meeting all inclusion criteria and no exclusion
criteria, assign it to one of the five categories. Only type the number of the group
as "1", "2", "3", "4" or "5" in your response. Do not type anything else.
"""
```

Listing 1: 5-Tier Approach Prompt

```
CAL_X_PROMPT = """You are an expert researcher rigorously screening titles and
abstracts of scientific papers for inclusion or exclusion in a review paper. Use
the criteria below to inform your decision. If any exclusion criteria are met or
not all inclusion criteria are met, exclude the article. If all inclusion
criteria are met, include the article. Only type "1" for include or "0" for
exclude to indicate your decision. Do not type anything else."""
```

Listing 2: Cal-X Prompt

```
# Since it's output is empty meaning all the files were modified
comm -3 \
  <(find client/src -type f \( -name "*.jsx" -o -name "*.tsx" \) | sort) \
  <(git diff c5aee8d01291e735bbadf04c8e431943f1a40a0e HEAD --name-only -- client/src
    | grep -E "\.jsx$|\.tsx$" | sort)
```

Listing 3: Calculating Modification Count

B Additional Tables

C Additional Figures

Table 2: Available AI Models and Their LLM Arena Ranking

Model ID	**Overall**	**Instruction Following**
deepseek-r1-distill-llama-70b *(not sure if same model)*	14	12
meta-llama/llama-4-maverick-17b-128e-instruct	52	52
meta-llama/llama-4-scout-17b-16e-instruct	-	-
moonshotai/kimi-k2-instruct	-	-
openai/gpt-oss-120b	36	27
openai/gpt-oss-20b	49	45
playai-tts	-	-
playai-tts-arabic	-	-
qwen/qwen3-32b	39	34

Table 3: Available AI Models and LiveBench Reasoning Average

Model ID	**Overall**	**Reasoning Average**
deepseek-r1-distill-llama-70b	-	59.81
meta-llama/llama-4-maverick-17b-128e-instruct	52	43.83
meta-llama/llama-4-scout-17b-16e-instruct	-	-
moonshotai/kimi-k2-instruct	-	62.97
openai/gpt-oss-120b	36	77.61
openai/gpt-oss-20b	49	-
playai-tts	-	-
playai-tts-arabic	-	-
qwen/qwen3-32b	39	83.08

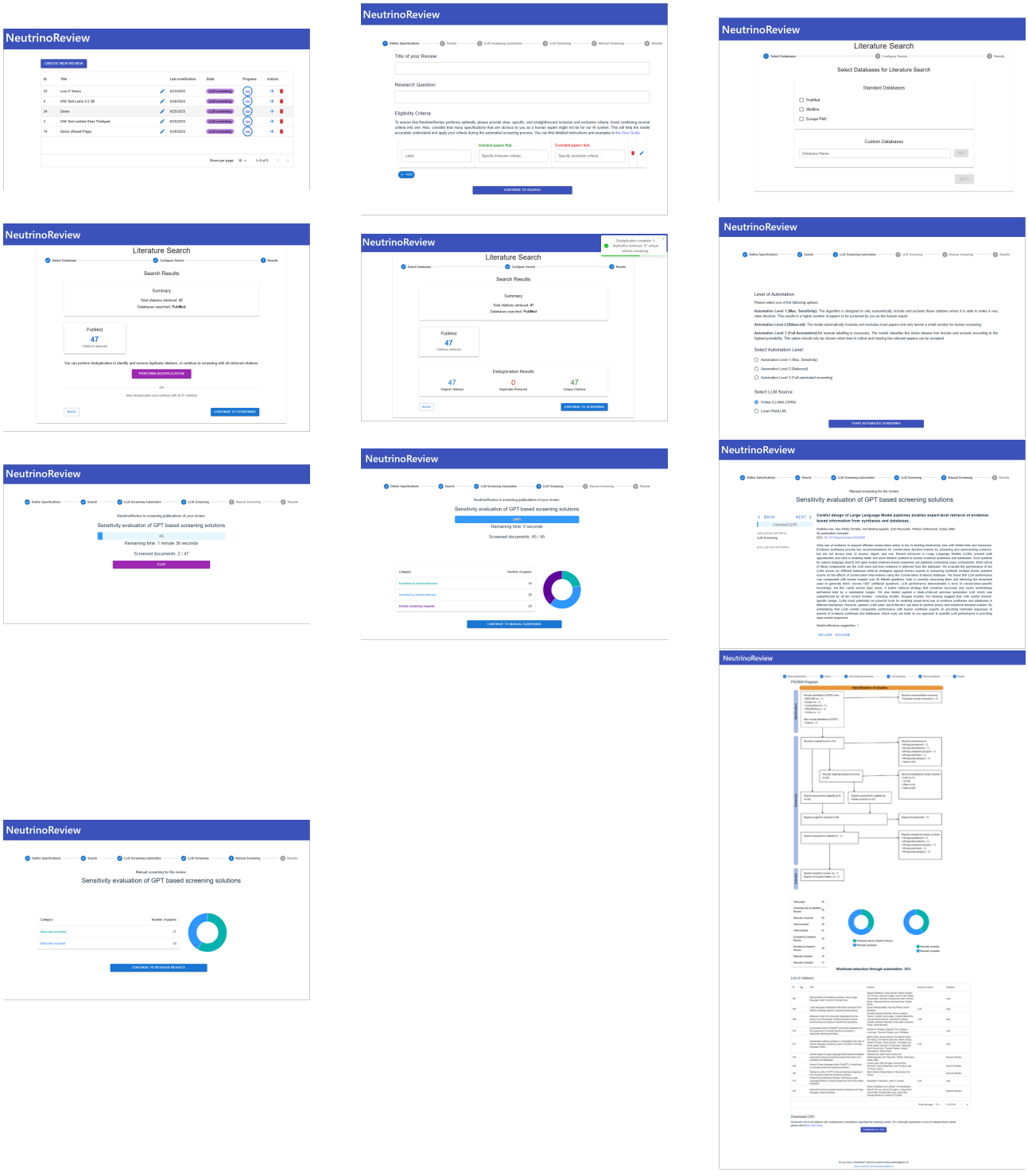


Figure 6: Screenshots of the Old UI.

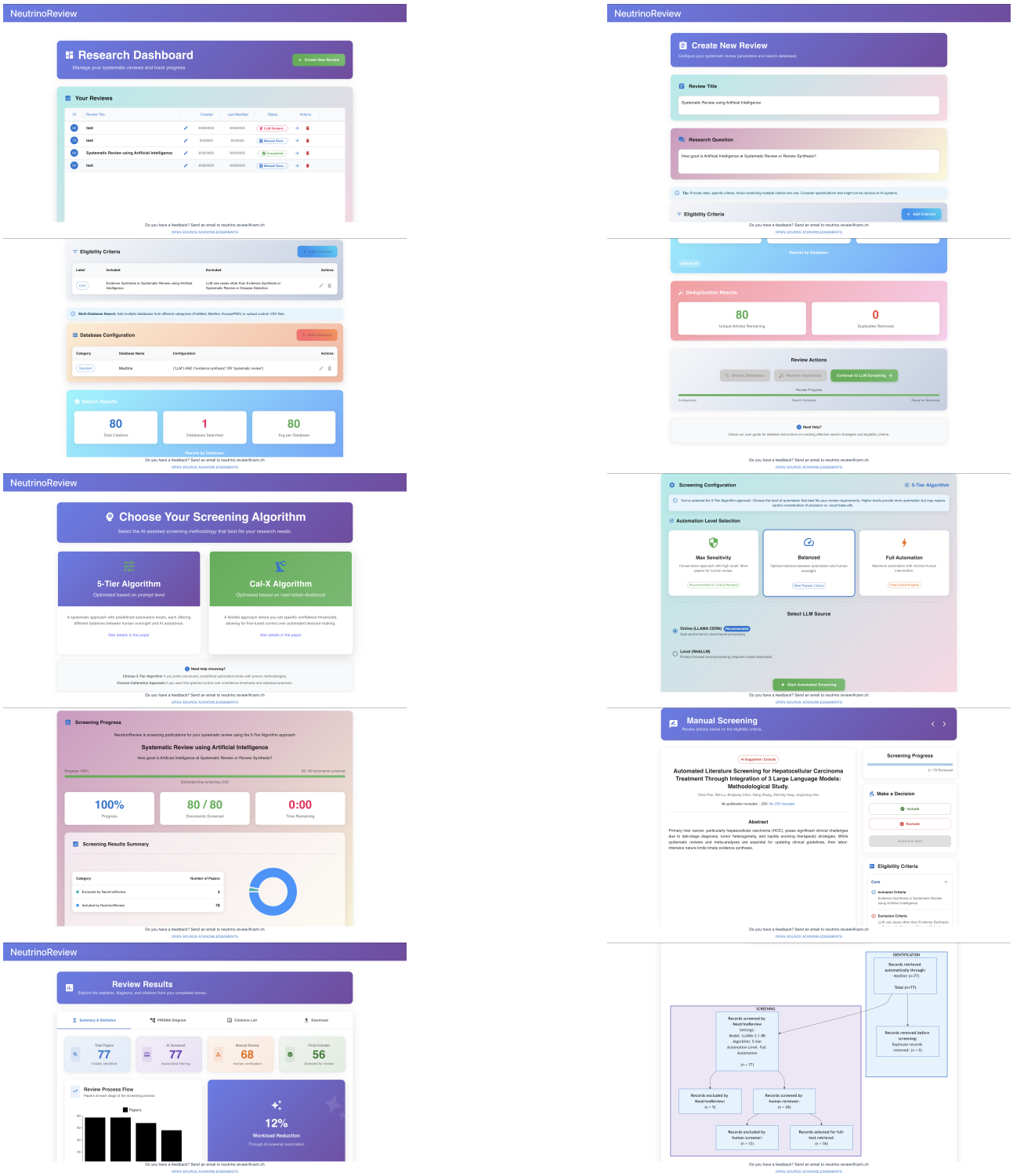


Figure 7: Screenshots of the New UI.