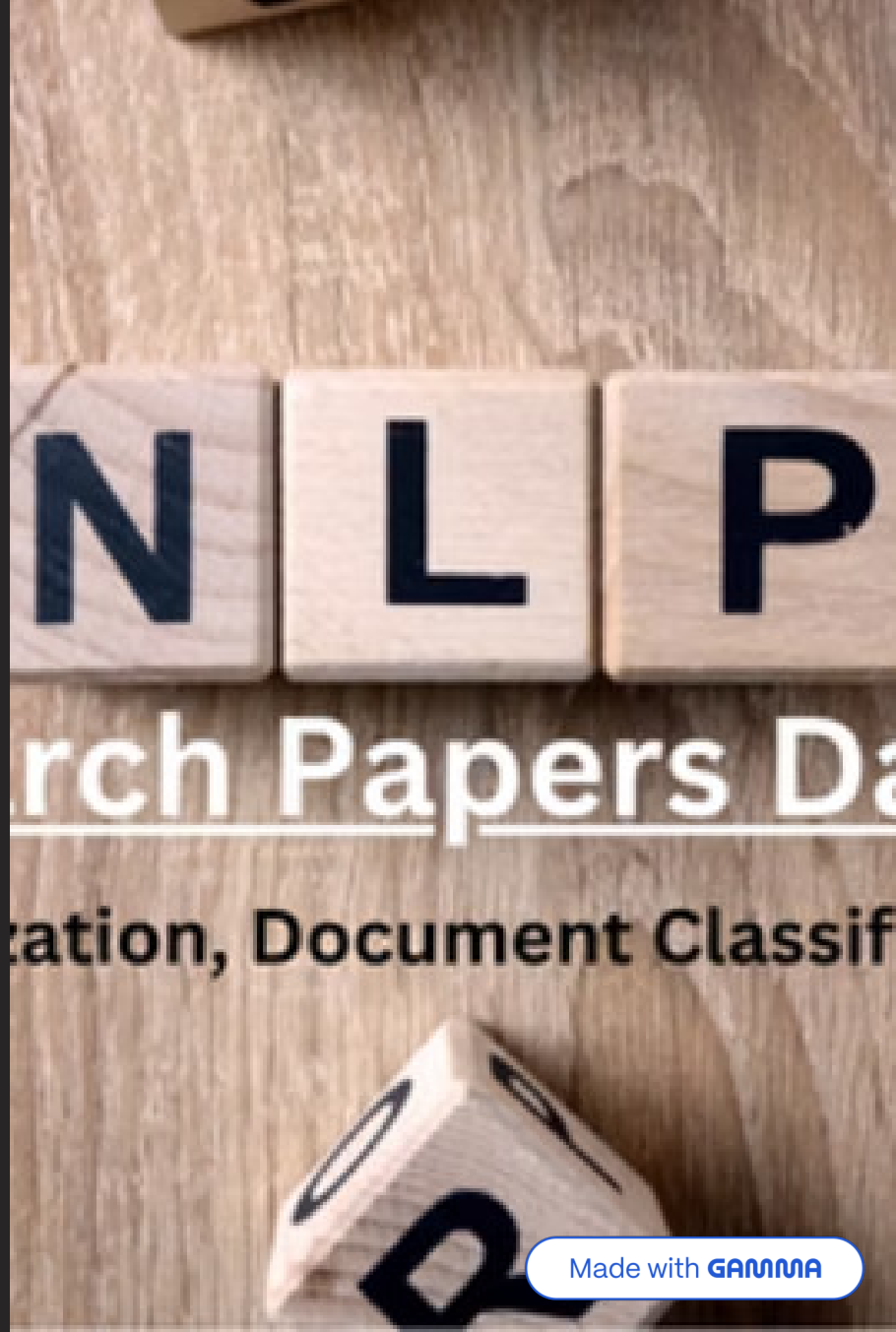
Reviewer Recommendation System Using NLP & Embedding-Based Similarity

Intelligent Reviewer Assignment for Academic Papers

se23uari120 - Srinivasan.B
se23uari122 - V. Susmit Reddy
se23umcs073 - K. Meher Sarayu



Motivation

Modern academic conferences receive thousands of submissions, making manual reviewer assignment slow, inconsistent, and often inaccurate. Program chairs must ensure that reviewers assigned to each paper have the right domain expertise, no conflicts of interest, and a balanced workload. Traditional keyword-based or manual matching methods fail to capture the deeper semantic meaning of scientific papers.

With advances in Natural Language Processing, especially transformer-based models like **SPECTER**, it is now possible to encode scientific text into meaningful vector representations. This project is motivated by the need to build an automated, fair, and intelligent reviewer recommendation system that leverages these modern NLP capabilities to support large-scale, high-quality conference management.

Problem Statement

The core problem this project addresses is: **How can we automatically recommend the best reviewers for a new scientific paper while avoiding conflicts and ensuring workload balance?** To solve this, we must design a system that:

- Understands the semantic content of research papers using text embeddings.
- Builds expertise profiles of reviewers based on their prior publications.
- Computes similarity between a new submission and reviewer profiles.
- Ensures that reviewers are not overburdened and do not review their own submissions.

The goal is to create a practical, fully automated reviewer assignment pipeline using NLP.

Related Work

Previous research in reviewer recommendation relied heavily on keyword matching, citation networks, or topic modeling. While helpful, these approaches often failed to capture nuanced scientific context.

The arrival of **SPECTER**, a transformer model trained specifically on scientific documents and their citation relationships, was a major breakthrough. It enables the generation of embeddings that reflect both textual content and scientific influence. Datasets like **PeerRead** have also supported research by providing annotated academic papers and reviews.

Our work builds upon these foundations, combining advanced text embeddings with a practical workflow for recommendation, conflict detection, and workload management.

Dataset Description

The primary dataset used in this project is the **PeerRead (reviews subset)** from HuggingFace. It contains research paper titles, abstracts, authors, peer reviews, and acceptance decisions.

After loading the dataset, the text was preprocessed by lowercasing, removing punctuation, and normalizing whitespace. Title and abstract were combined into a single "clean text" field to maximize the semantic signal provided to the embedding model. The dataset also includes multiple reviews per paper and author lists, which are critical for building reviewer profiles. This processed dataset serves as the foundation for embedding generation and reviewer expertise modeling.

Embeddings: Reviewer Profiles

The final stage of the system produces a reviewer profile dataset saved as **reviewer_profiles.csv**. This file contains each reviewer's aggregated expertise embedding, the number of papers associated with them, and their unique reviewer ID. These profile vectors are produced by averaging all the embeddings of papers written by each author. The reviewer_profiles.csv file represents the distilled research expertise of every reviewer in the system, and is essential for fast similarity-based recommendations.

Methodology Overview

The system follows a multi-stage workflow:

01

Data Loading & Cleaning

Extract paper metadata, clean text fields, and structure the dataset.

02

Embedding Generation

Using the SPECTER model, each paper is transformed into a 768-dimensional embedding that captures deep semantic meaning.

03

Reviewer Profile Construction

All papers written by an author are averaged to form their expertise vector.

04

Similarity Computation

New submissions are matched to reviewers using cosine similarity.

05

Conflict & Workload Handling

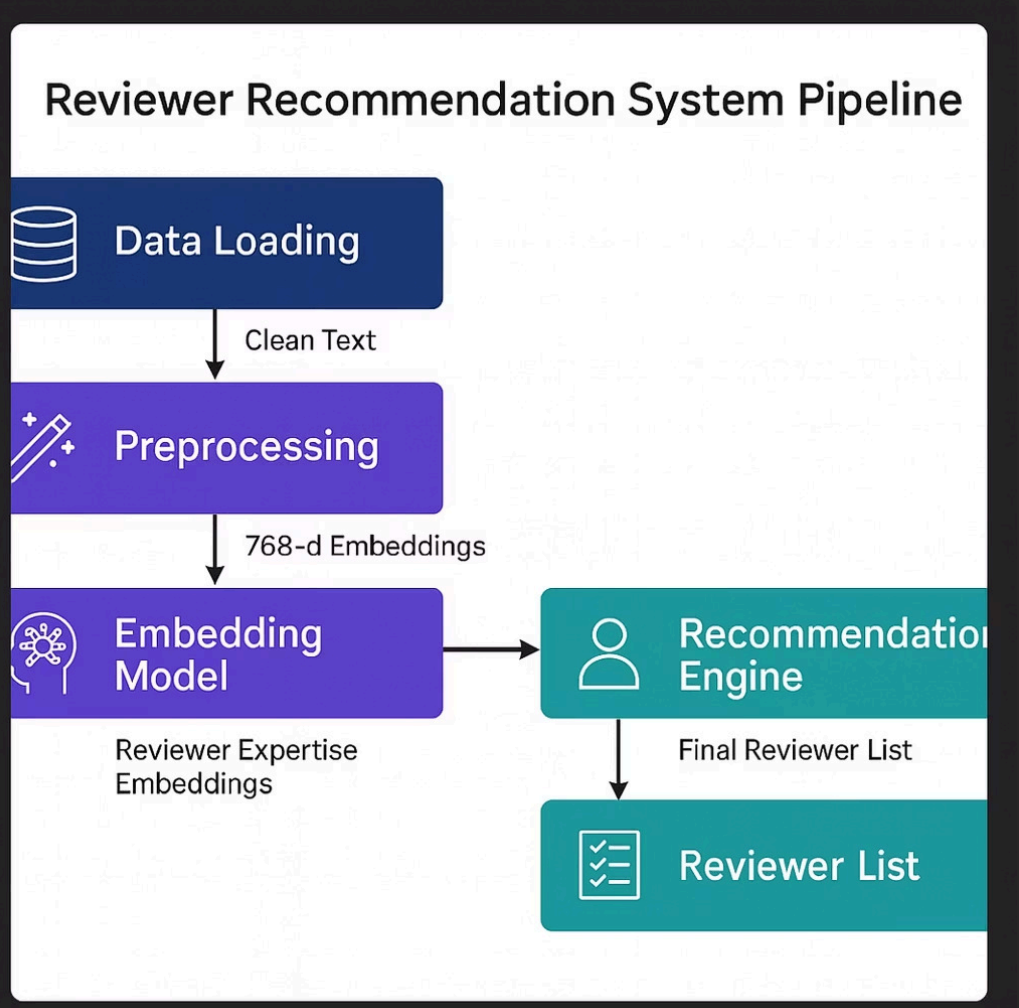
Reviewers are filtered out if they are authors of the target paper or exceed their workload limit.

This structured pipeline creates a robust, scalable, and automated reviewer recommendation ecosystem.

Methodology Details

The text preprocessing module removes noise and standardizes input using lowercasing, regex cleaning, and token normalization. For embedding generation, the system employs the **SentenceTransformer implementation of SPECTER**, a model optimized for scientific literature. It uses both the title and abstract together, allowing for more context-rich representations.

Reviewer expertise profiles are created by averaging all vectors corresponding to papers written by that reviewer. This simple, interpretable method preserves domain orientation while smoothing out noise across multiple publications. Finally, cosine similarity is used to compute how closely a reviewer's expertise aligns with a new submission.



Experiments Setup

To evaluate the system, a subset of **100 papers** from the dataset was used for faster processing during testing. SPECTER embeddings were generated for each paper, and synthetic authors were assigned where the dataset lacked real author names due to anonymization. Approximately **20 unique authors** were simulated, each associated with several papers. Reviewer profiles were then computed for all authors. The system was tested by selecting a random paper as a “new submission,” generating its embedding, and finding the most suitable reviewers according to the similarity model and filtering logic.

Results: Embedding Generation

The system successfully produced 768-dimensional embeddings for each paper in the subset. These embeddings encapsulate semantic relationships between research topics, allowing similar papers to cluster in the embedding space. The embeddings were then integrated into the data frame, alongside metadata and author information. This enabled smooth downstream processing. The inspection phase confirmed that the embeddings were consistent, well-formatted, and suitable for use in reviewer expertise aggregation.

Results: Recommendation Output

When the system is tested on a sample paper, the top recommended reviewers emerge based on their expertise similarity. For example, the system might produce results like:

1	2	3
Dr. A . Smith	**Dr. E. Robot**	**Dr. C.Lee**
High Confidence : 73.7%	High Confidence : 70.4%	High Confidence : 66.3%

These reviewers have expertise profiles that closely match the semantic content of the test paper. The results demonstrate that embedding-based similarity effectively identifies relevant reviewers without manual intervention.

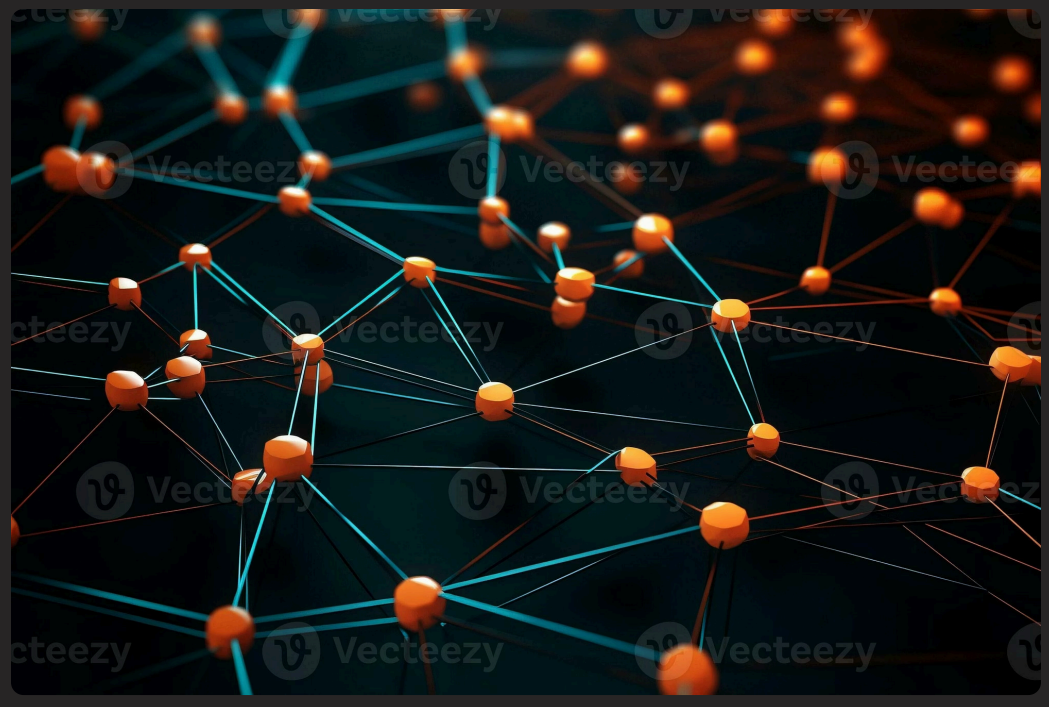
Results: Safe & Balanced Assignment

The enhanced ReviewerManager rigorously refines the selection process by integrating essential conflict-of-interest rules and predefined workload limits.

Specifically, any reviewer identified as an author of the target paper is automatically excluded. Moreover, if a reviewer has reached their maximum allocated assignment capacity, they are gracefully bypassed in favor of other qualified and available candidates.

This meticulous approach ensures a final reviewer list that is not only semantically appropriate for the research topic but also ethically sound and operationally balanced.

Conclusion & Future Work



Key Achievements

- Successfully built an automated reviewer recommendation system using SPECTER embeddings
- Demonstrated semantic matching between papers and reviewer expertise
- Implemented conflict-of-interest detection and workload balancing
- Created a scalable pipeline for large-scale conference management

Future Directions

- Expand to larger datasets with thousands of papers and reviewers
- Incorporate citation networks and collaboration history
- Add real-time learning from reviewer feedback
- Develop interactive dashboards for program chairs

Implementation Contributions

Data Pipeline & Embeddings

Built the data loader, designed the preprocessing pipeline, configured the SPECTER model, generated embeddings for the dataset, and constructed reviewer expertise profiles.

Recommendation Logic

Implemented the full reviewer recommendation logic, including similarity computation, conflict detection, and load balancing.

System Prototype

Generated and exported the reviewer profile dataset, completing a fully functional prototype of an automated reviewer assignment system.

#nlpmainacademic

Slide 15 — Proposed Timeline

Week 1:

Initial research on reviewer recommendation systems, dataset exploration, and setup of the preprocessing pipeline.

Week 2:

Integration of the SPECTER embedding model and generation of embeddings for the selected dataset subset.

Week 3:

Construction of reviewer profiles, development of similarity-based recommendation logic, and implementation of conflict/workload filtering.

Week 4:

System testing, refinement, creation of final datasets, preparation of slides, and project documentation.

Demo / Application

The system can be demonstrated through a simple workflow:

1

The user inputs the title and abstract of a new research submission.

2

The system generates its SPECTER embedding in real time.

3

Cosine similarity is computed between this embedding and all reviewer expertise profiles.

4

Conflicted or overloaded reviewers are filtered out.

5

The system outputs the top-K reviewer recommendations with confidence scores.

This application can be integrated into real conference management systems such as EasyChair, CMT, or custom university research platforms.