Autonomous Machine Learning-Based Peer Reviewer Selection System

Nurmukhammed Aitymbetov, Dimitrios Zorbas
Nazarbayev University, School of Engineering & Digital Sciences, Astana, Kazakhstan
E-mail: {firstname.lastname}@nu.edu.kz

Abstract

The peer review process is essential for academic research, yet it faces challenges such as inefficiencies, biases, and limited access to qualified reviewers. This paper introduces an autonomous peer reviewer selection system that employs the Natural Language Processing (NLP) model to match submitted papers with expert reviewers independently of traditional journals and conferences. Our model performs competitively in comparison with the transformer-based state-of-the-art models while being 10 times faster at inference and 7 times smaller, which makes our platform highly scalable. Additionally, with our paper-reviewer matching model being trained on scientific papers from various academic fields, our system allows scholars from different backgrounds to benefit from this automation.

1 Introduction

Peer review is a major component of the academic publishing process that ensures the integrity and quality of scholarly research. Traditionally, peer review has been a manual and often cumbersome process, hindered by prolonged turnaround times. With the growing volume of paper submissions, manual reviewer selection has become impractical, leading to the development of automated paper-reviewer matching algorithms. However, even with these advancements, authors still face challenges. They are often required to adhere to strict deadlines and wait extended periods for their reviews, leaving them with limited time to respond to peer reviewers' feedback (Huisman and Smits, 2017). This can impede the feedback loop and limit authors' opportunities to improve their work. Furthermore, submitting to specific journals or conferences may introduce conflicts of interest and biases in feedback, as the process can be manipulated to favor or hinder certain submissions (Adler and Stayer, 2017; da Silva et al., 2019).

In response to these issues, we introduce an autonomous peer reviewer selection system designed to function independently of the traditional academic publishing venues, such as conferences and journals. It could be used to provide a flexible and efficient alternative for researchers seeking preliminary review of their papers. This system facilitates rapid reviewer assignment, enabling researchers to receive timely feedback. Importantly, it operates continuously, allowing users to submit their papers at any time without being constrained by traditional deadlines. By decoupling the peer review process from the traditional venues, the platform also aims to minimize the potential for conflicts of interest and biases, as reviewers are less likely to be influenced by the stakes of formal decision-making. The proposed platform aims to democratize and accelerate the peer review process, offering researchers the opportunity to improve their work before formal submission to journals or conferences. By facilitating quick and high-quality peer reviewer assignments from a global pool of experts, the system has the potential to enhance the overall quality of academic publications. Its scalability and efficiency also make it a promising solution for the future, with the capability to support a large database of authors and reviewers.

The core innovation of the system lies in its custom paper-reviewer matching model, which is significantly smaller and faster than existing transformer-based models while maintaining competitive performance. This efficiency allows the system to scale effectively, accommodating the needs of a potentially large number of users without compromising the quality of the reviewer matches. Moreover, unlike many automated matching systems that are typically developed and fine-tuned for specific fields such as computer science or machine learning, the proposed model is trained on a diverse set of academic disciplines. This makes the platform accessible to scholars from a wide range

Proceedings of the 31st International Conference on Computational Linguistics: System Demonstrations, pages 199–207
January 19–24, 2025.

of fields, ensuring that they can also benefit from rapid and high-quality reviewer assignments.

Overall, the contributions of this work are summarized as follows:

- We develop and implement an open-source prototype of a peer review system that operates independently of traditional journals and conferences, featuring continuous paper submission and automated reviewer assignment[1].
- We introduce an efficient GRU-based paper-reviewer matching model that performs comparably to existing transformer-based approaches, while being significantly smaller and faster at inference.
- We show that classification-based pre-training using subject-area classification can be effective for learning paper representation vectors useful for paper-reviewer matching task. The learned representation vectors capture meaningful topic information and measure paper-reviewer affinity surprisingly well.

The rest of the paper is organized as follows. Section 2 provides an overview of the related research on the paper-reviewer matching problem and the current systems used in practice. Section 3 delves deeper into the technical description of the proposed system, including the details of our paper-reviewer matching model. Section 4 describes the experimental setup used to evaluate the performance of our paper-reviewer matching system. Section 5 concludes the paper by summarizing key findings and offering suggestions for further improvements. Section 6 discusses some limitations of the proposed system.

2 Related Research

The use of automatic paper-reviewer matching systems is not a new trend in the academic world and have been studied for almost a decade (Li and Watanabe, 2013). Modern paper-reviewer assignment systems mainly consist of three components: (1) expertise modeling system, (2) reviewer assignment system and (3) conflict-of-interests (COI) detection system. The first component involves the development of models that accurately represent whether the reviewer has the required topical expertise to review the submitted paper. The second component involves actually assigning reviewers to papers based on the expertise modeling results. The third component involves detecting any relationship reviewers and authors may have and addressing them in order to ensure fair review.

2.1 Expertise modeling

Expertise modeling is essential for aligning papers with reviewers who possess relevant knowledge. Initial approaches in this area relied on keyword matching (Conry et al., 2009) and simple word-based techniques such as TF-IDF to measure similarity between paper content and reviewers' past publications (Yarowsky and Florian, 1999; Hettich and Pazzani, 2006). More advanced approaches introduced topic modeling methods such as Latent Dirichlet Allocation (LDA), which generates topic distributions for both papers and reviewers to calculate a more abstract similarity (Mimno and McCallum, 2007). These models have been widely adopted in conference management systems such as the Toronto Paper Matching System (TPMS) (Charlin and Zemel, 2013) and IEEE INFOCOM Reviewer Assignment System (Li and Hou, 2016).

A more recent approach to expert modeling is based on neural network models, which represent papers and reviewers as dense vectors (document embeddings). These models capture deeper semantic features, making them highly effective for paper-reviewer matching. In particular, scientific paper representation models, such as SciBERT (Beltagy et al., 2019), SPECTER (Cohan et al., 2020), and SciNCL (Ostendorff et al., 2022), have become prominent in the field, and they have been adopted by OpenReview, a platform widely used in major conferences such as NeurIPS and ICLR (OpenReview, 2024). These models represent both papers and reviewers as vector embeddings, allowing for the computation of similarity between them. The similarity score reflects the reviewer's expertise relative to the paper's topic, which can be used to assign the best-suited reviewers. Our system adopts this document embedding approach for representing papers and reviewers' profiles, leveraging these embeddings to compute expertise scores.

2.2 Reviewer assignment

In traditional systems, reviewer assignment is often handled via matching-based approaches, where all papers and reviewers are considered simulta-

[1]The source code and a video demonstration is found in this link: https://github.com/nurmybtw/autonomous-peer-review-platform

neously, and the assignment is determined using optimization algorithms such as Integer Linear Programming or Mixed Integer Programming (Charlin and Zemel, 2013; Leyton-Brown et al., 2022). This optimization is primarily used in batch-processing scenarios, such as conferences that collect all submissions before a deadline and then process them in bulk, matching papers to a set of reviewers. In contrast, our system follows a retrieval-based approach, where papers are served on a rolling basis and assigned to reviewers individually. As an online system, our platform continuously matches papers with the best available reviewers based on their expertise.

2.3 COI detection

Most of the traditional peer review systems also implement COI detection system to minimize the biases (Tang et al., 2010; Wu et al., 2018; Leyton-Brown et al., 2022). However, our system does not incorporate COI detection, as its primary goal is to provide independent feedback rather than formal acceptance into a journal or conference. We prioritize reviewer expertise over potential conflicts, ensuring that authors receive high-quality feedback without being constrained by COI limitations.

3 Proposed System

In this section, we present our system and the paper-reviewer matching model within it. First, we describe how the system operates on a high-level. Then, we describe core technical innovation: efficient paper-reviewer matching model based on bidirectional Gated Recurrent Unit (GRU) and classification-based pre-training.

3.1 System Overview

The core of the platform is a two-part system that analyzes submitted paper abstracts to find the best matching reviewers. It is important to note that all the components within this system focus only on the papers' content and do not consider any identifying information related to the author or potential reviewer. This minimizes biases, while maximizing the quality of matches.

The information flow begins with authors submitting their research papers via the web interface, providing the title and abstract along with the document. As depicted in Figure 1, the proposed system employs a sequential pipeline comprising two interconnected components: topic-based filtering and expertise-based ranking.

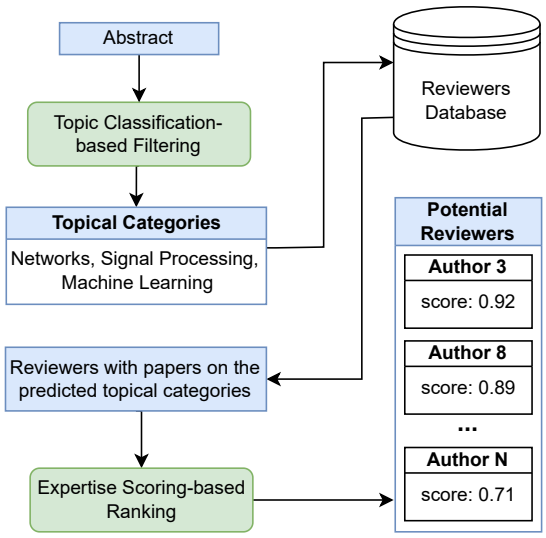

Figure 1: Proposed system flow consisting of two major components: initial filtering based on topic classification and final ranking based on expertise scoring.

The first component of the system involves the classification of the incoming paper abstract into three of the 158 predefined topic domains. After determining the paper topics, only the reviewers who have written a certain amount of papers on those topics are selected for the next step. This greatly reduces the number of reviewers and papers to be analyzed in detail during the expertise scoring process, optimizing the response time of the overall system, as the evaluation of the textual data of possibly thousands of authors is a costly operation. Choosing three categories for each paper abstract reflects the possibility of the paper belonging to multiple topics.

Then, the system transitions to the second stage, where the potential reviewers are assigned expertise scores and ranked accordingly; a higher score represents a closer match between the potential reviewer's expertise and the paper topics. This scoring process is based on cosine similarity, comparing the latent representation of the incoming paper's abstract with candidate reviewers' past publications. This method facilitates fine-grained ranking of reviewers.

The choice of combining topic-based filtering with expertise-based ranking stems from the need to balance efficiency and precision. Topic-based filtering serves as a rapid initial filter, eliminating clearly unqualified reviewers from the pool. Subsequently, expertise-based ranking using the latent representations offers a more detailed and nuanced assessment of each reviewer's expertise, ensuring

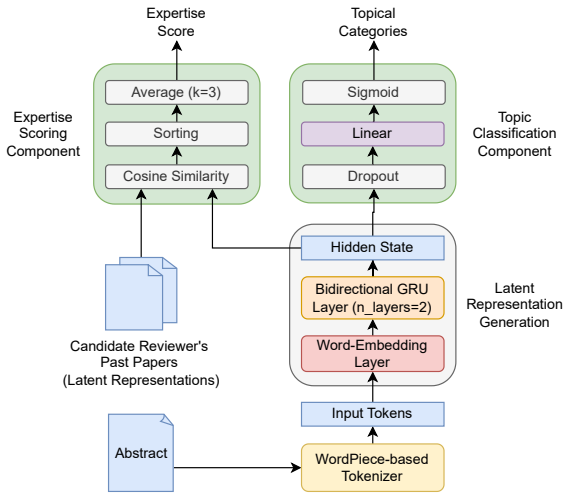

Figure 2: Model Architecture. The model is used for both topic classification and expertise scoring components. The Second GRU layer's hidden state is used as a latent representation vector for papers.

that the final matches are highly relevant.

Finally, after ranking the potential reviewers based on their expertise scores, two of the best available candidates are selected and sent a review request. If the selected reviewer accepts the review request, this reviewer will be officially assigned to the paper for providing feedback. Authors can then utilize this feedback to improve their manuscripts before submitting them to journals or conferences. In case of rejection, a review request is sent to the next available candidate. If the potential reviewer does not respond to the request for a certain amount of time, authors will have the opportunity to initiate the search for a new reviewer via the platform.

3.2 Model

The core of our paper-reviewer matching system is a Gated Recurrent Unit (GRU)-based model that serves a dual purpose: predicting the topical categories of submitted papers and generating latent representations of the abstracts for subsequent expertise scoring (see Figure 2). This dual functionality is achieved using a classification-based pre-training approach that we describe in the next paragraph.

Classification-based pre-training Zhang et al. (2020) interpreted paper-reviewer matching as a multi-label classification task. In their approach, the model was first trained to generate representation vectors of abstracts, which were then used for multi-label classification. The matching was based on the degree of alignment between the predicted labels of the submitted paper and the predicted labels of the reviewer's past papers, demonstrating the effectiveness of classification-based methods for this task. In contrast, our approach first pre-trains the GRU model for multi-label classification, then uses the learned representation vectors for matching. Evaluations demonstrate that GRU's hidden state can be surprisingly effective when used as a representation vector of abstracts. The model's ability to generate useful latent representations for paper-reviewer expertise modeling while being trained primarily for classification might be logical and intuitive. When assessing a reviewer's suitability, one naturally examines their research areas. Thus, a model trained to classify topics inherently learns to generate representation vectors that encapsulate these research areas effectively.

Expertise scoring To compute the expertise score, the system first retrieves the latent representations of the abstracts from the most recent papers authored by each reviewer r_i. Let $\mathbf{p}_i^1, \mathbf{p}_i^2, \ldots, \mathbf{p}_i^m$ represent the representation vectors of the m papers authored by r_i, and let $\mathbf{s}$ denote the representation vector of the submitted paper. Following Stelmakh et al. (2023), we limit m to the last 10 papers published by each reviewer, as they showed that using more than 10 papers provides minimal additional benefit. The system then computes the cosine similarity $\cos(\mathbf{s}, \mathbf{p}_i^j)$ between the submitted paper's representation vector $\mathbf{s}$ and each representation vector $\mathbf{p}_i^j$ of the reviewer's previous papers. Notably, we use only the abstracts of these papers to obtain latent representations, rather than the full text, as Stelmakh et al. (2023) demonstrated that the performance difference between using abstracts and full text is marginal.

For each reviewer r_i, we consider the top three cosine similarity scores, which correspond to the three most related papers the reviewer has authored. The final expertise score E_i for reviewer r_i is computed as the average of these top three scores: $E_i = \frac{1}{3}\sum_{k=1}^{3} \cos(\mathbf{s}, \mathbf{p}_i^k)$. The reason for averaging the top three scores stems from our assumption that reviewers have good expertise for reviewing the submitted paper if they have written at least three related papers. Authors often focus on different topics, and using the top three scores provides a robust measure of a reviewer's expertise. If a reviewer has written fewer than three related papers, their overall score will automatically reduce as a result.

Training The model was trained to classify topics on a subset of the open-source ArXiv dataset from Kaggle (Cornell University, 2020) for 20 epochs with a batch size of 256 using NVIDIA P100 GPU. It contains metadata such as title, abstract, authors, and topical categories for 2.4M academic papers featured in the ArXiv repository. The dataset includes papers spanning 158 categories across 8 academic fields included in ArXiv's official categorical taxonomy. The dataset provides rich categorical labels for each entry, allowing for multi-label classification. For detailed view of these categories, refer to Appendix A.

Dataset pre-processing The ArXiv dataset (Cornell University, 2020) originally contained approximately 2.4 million entries with varying numbers of categories assigned to each entry. Most entries had only one label, followed by those with two and three labels. Instances with more than three labels were much less common. For the purpose of training the model, only multi-label entries with two or three categories were used, as these provide richer context for expertise scoring. This selection process was designed to focus on multi-label instances because papers can naturally belong to multiple fields, offering more informative training data. To address class imbalance issue in the dataset, a cap of 15,000 entries per category was applied, resulting in a balanced subset of 840K entries, 710K of which were used for training. The first category assigned to each paper was used for random stratified splitting into training and test sets; however, all assigned categories were used during training.

Tokenization A custom WordPiece-based tokenizer (Wu et al., 2016) was trained on the ArXiv dataset's training set, resulting in a vocabulary of 50,000 tokens. The tokenizer was implemented using the HuggingFace's BertTokenizer class. The model was configured to accept inputs with a maximum length of 256 tokens, with both padding and truncation applied.

4 Evaluation & Discussion of the results

In this section, we present the evaluation details of our paper-reviewer matching model. First, we start off by defining the experimental settings in terms of the metrics and datasets used in our evaluation. Then, we briefly describe the baseline and state-of-the-art models used for comparison. Finally, we present the results and discuss certain implications.

4.1 Evaluation Datasets

For evaluating topic classification performance, we employed the test set from the previously mentioned arXiv dataset (Cornell University, 2020). Refer to Section 3 for dataset details.

For expertise scoring, we utilized the dataset presented by Stelmakh et al. (2023). OpenReview platform uses this dataset to evaluate its models (OpenReview, 2024), making it an ideal fit for our tests. It contains 477 self-reported expertise scores from 58 researchers evaluating papers they have read recently. Each researcher rated their expertise for a given paper on a scale from 1.0 (not qualified) to 5.0 (fully qualified). These evaluations cover both easy (large difference in expertise scores) and hard (small difference in expertise scores) cases. The dataset is well-suited for evaluating expertise scoring models, with participants' profiles constructed from up to 20 of their most recent publications with titles and abstracts included.

4.2 Evaluation Metrics

For topic classification, we used two metrics: **Single-match Accuracy**, which measures the percentage of cases where at least one of the three predicted topics matches a true topic of the paper, and **Recall@3**, which calculates the proportion of true topics that appear in the top three predicted topics for each paper.

The expertise scoring was evaluated using metrics defined by Stelmakh et al. (2023). The primary metric is a **Loss** based on a modified Kendall's Tau distance, penalizing incorrect ranking of paper pairs by the difference in their true expertise scores. Also, **Easy Triplets Accuracy** and **Hard Triplets Accuracy** are measured as the fraction of correctly ordered paper pairs in terms of researcher's predicted expertise for large differences (easy triplets) and small differences (hard triplets) in true expertise scores, respectively. Lower loss and higher accuracy across triplet categories indicate better performance.

Finally, for model efficiency, we evaluated the system's **Inference Time per 1000 Samples**, and **Model Size** in terms of the number of parameters in the model. Efficiency was evaluated using NVIDIA P100 GPU.

4.3 Comparison Models

We compare our model with scientific representation models featured in OpenReview platform:

SciBERT (Beltagy et al., 2019), SPECTER2 (Singh et al., 2023), SciNCL (Ostendorff et al., 2022). For topic classification comparison, SciBERT and SciNCL were fine-tuned for 2 epochs in a multi-label classification setting. For SPECTER2, since it is an adapter-based model, we fine-tuned it by training a new adapter in a multi-label classification setting for 2 epochs. For expertise scoring comparison, we used the base versions of the models.

SciBERT SciBERT (Beltagy et al., 2019) is a pretrained language model specifically designed for scientific text. It is based on the BERT (Devlin et al., 2019) architecture but trained on a large corpus of scientific papers from the computer science and biomedical domains. SciBERT uses an in-domain vocabulary, making it more effective at processing scientific language compared to general-domain models such as BERT. SciBERT serves as the foundation for many state-of-the-art scientific document representation models, making it an important baseline.

SPECTER2 Building upon SciBERT, Cohan et al. (2020) introduced SPECTER. The key innovation in SPECTER is its use of the citation graph for learning document representations. It leverages contrastive learning by considering papers that cite each other as close in the embedding space, while papers without citation links are placed further apart. This approach improves performance on various document-level tasks such as recommendation and classification. SPECTER2 (Singh et al., 2023) extends this model by introducing task-specific adapters, for tasks such as proximity or regression. Additionally, it uses a larger and more diverse training set, which includes papers from a broader range of scientific fields, further enhancing its robustness across disciplines.

SciNCL SciNCL (Ostendorff et al., 2022) builds upon the idea used in original SPECTER (Cohan et al., 2020), which leverages citation graphs to inform contrastive learning samples. However, unlike SPECTER, which uses a discrete binary relationship (i.e., either papers cite each other or they do not), SciNCL employs a continuous similarity measure to capture more nuanced relationships between papers. It enhances contrastive learning by sampling positive examples not just from directly cited papers, but also from closely related papers within the k-nearest neighbors of the citation graph.

4.4 Results and Discussion

Tables 1 and 2 present the evaluation results of the proposed GRU-based model against state-of-the-art models in the paper-reviewer-expertise modeling field across both topic classification and expertise scoring (modeling) tasks.

Model	Single-match Accuracy (k=3)	Recall@3
SciBERT (fine-tuned)	96.69	0.794
SPECTER2 (fine-tuned)	95.95	0.771
SciNCL (fine-tuned)	96.51	0.789
Our model	95.32	0.766

Table 1: Performance comparison of different models on topic classification

Model	Loss	Easy Triplets	Hard Triplets
SciBERT (base)	0.30	0.82	0.55
SPECTER2 (base)	0.22	0.89	0.61
SciNCL (base)	0.22	0.91	0.65
Our model	0.26	0.83	0.57

Table 2: Performance comparison of different models on expertise scoring

Our GRU-based model, although slightly outperformed by more complex transformer-based models, demonstrates respectable performance in both tasks. In topic classification, it achieved a single-match accuracy of 95.32% and a Recall@3 of 0.766. In expertise scoring, our GRU-based model achieved a loss of 0.26. The model's accuracy on easy triples was 0.83, and on hard triples, it was 0.57. While these metrics are lower than those of the state-of-the-art transformer-based models (SPECTER2 and SciNCL), our model outperformed SciBERT baseline in expertise scoring.

Despite being mostly inferior to the state-of-the-art models in both tasks, our model offers significant efficiency gains (see Table 3). Our model has a significantly faster inference time (around 1.7 seconds per 1000 samples) compared to the transformer-based models, which require around 16 to 18 seconds. Moreover, our model is much smaller with 15M parameters, compared to 110M parameters of the BERT-based models.

This efficiency makes our model highly suitable for large-scale systems like ours, where thousands of scholars may use the platform. This trade-off between performance and efficiency is critical for the proposed system, ensuring rapid and scalable processing without compromising the overall quality

Model	Model Size (params)	Inference time per 1000 samples (seconds)
SciBERT	110M	16.62
SPECTER2	111M	18.22
SciNCL	110M	16.87
Our model	15M	1.72

Table 3: Efficiency comparison of different models in terms of model size and inference time spent per 1000 samples

of the automation.

Interestingly, in the expertise modeling task, the GRU-based model performs surprisingly well. This result suggests that pre-training models on topic classification can be effective for paper-reviewer expertise modeling. The explanation for this behavior might be intuitive, since topic classification requires the model to learn latent representations that encapsulate the topical areas, which are naturally used for assessing the relevance of a reviewer's expertise. We suggest the further exploration of classification-based pre-training to understand the potential of this approach in paper-reviewer-expertise modeling and more general task of scientific text representation.

5 Conclusion

Our research presented a prototype of an autonomous peer reviewer selection system that effectively leverages NLP techniques to streamline the peer review process. By employing a GRU-based model, our system demonstrates a solid balance between accuracy and efficiency. The continuous and on-demand nature of the system offers researchers rapid access to expert feedback, bypassing the constraints of traditional review cycles tied to specific journals or conferences.

A key area for future improvement is the development of an effective reviewer onboarding system, which is essential for ensuring the platform has a high-quality pool of reviewers. Furthermore, it is important to integrate a feedback mechanism where users can rate the quality of reviewer matches and the usefulness of the feedback. These ratings could be used to iteratively adjust and improve the matching system.

Additionally, we suggest further development of the classification-based pre-training, as it shows a potential in paper-reviewer matching and the broader field of scientific text representation.

6 Limitations

While the proposed system demonstrates significant potential, several limitations remain, both at the platform and model levels. The key challenge lies in recruiting and motivating reviewers to visit the platform and perform "out-of-formal" reviews. Since these reviews are independent of traditional academic venues like journals and conferences, encouraging expert reviewers to join and contribute actively remains a limiting factor. A potential solution could involve a system where users must contribute reviews to receive feedback on their own submissions.

Additionally, the current paper-reviewer matching model supports 158 categories across 8 academic fields, which may not capture the full granularity of many specialized fields. However, it should be noted that our model performs well even with this limited number of categories, suggesting that using more fine-grained taxonomies could further improve the model's performance and adaptability to niche topics.

Acknowledgments

This publication has emanated from research conducted with the financial support of the Ministry of Science and Higher Education of the Republic of Kazakhstan for the project "Leveraging IoT Mesh Networks for Machine Learning Knowledge Transfer" (grant No. AP23487072).

References

Adam C. Adler and Stephen A. Stayer. 2017. Bias among peer reviewers. *JAMA*, 318 8:755.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Laurent Charlin and Richard Zemel. 2013. The toronto paper matching system: an automated paper-reviewer assignment system. In *Proceedings of the 2013 ICML workshop on peer reviewing and publishing models*.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association*

for Computational Linguistics, pages 2270–2282, Online. Association for Computational Linguistics.

Don Conry, Yehuda Koren, and Naren Ramakrishnan. 2009. Recommender systems for the conference paper assignment problem. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, page 357–360, New York, NY, USA. Association for Computing Machinery.

Cornell University. 2020. arxiv dataset. Accessed: 2024-09-29.

Jaime A. Teixeira da Silva, Judit Dobránszki, Radha Holla Bhar, and Charles T. Mehlman. 2019. Editors should declare conflicts of interest. *Journal of Bioethical Inquiry*, 16:279 – 298.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Seth Hettich and Michael J. Pazzani. 2006. Mining for proposal reviewers: lessons learned at the national science foundation. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 862–871, New York, NY, USA. Association for Computing Machinery.

Janine Huisman and Jeroen Smits. 2017. Duration and quality of the peer review process: the author's perspective. *Scientometrics*, 113(1):633–650.

Kevin Leyton-Brown, Yatin Nandwani, Hedayat Zarkoob, Chris Cameron, Neil Newman, Dinesh Raghu, et al. 2022. Matching papers and reviewers at large conferences. *arXiv preprint arXiv:2202.12273*.

Baochun Li and Y. Thomas Hou. 2016. The new automated ieee infocom review assignment system. *IEEE Network*, 30(5):18–24.

Xinlian Li and Toyohide Watanabe. 2013. Automatic paper-to-reviewer assignment, based on the matching degree of the reviewers. *Procedia Computer Science*, 22:633–642.

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*, pages 500–509.

OpenReview. 2024. Paper-reviewer affinity modeling for openreview. `https://github.com/openreview/openreview-expertise`. Accessed: 2024-09-24.

Malte Ostendorff, Nils Rethmeier, Isabelle Augenstein, Bela Gipp, and Georg Rehm. 2022. Neighborhood contrastive learning for scientific document representations with citation embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11670–11688, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2023. SciRepEval: A multi-format benchmark for scientific document representations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5548–5566, Singapore. Association for Computational Linguistics.

Ivan Stelmakh, John Wieting, Graham Neubig, and Nihar B. Shah. 2023. A gold standard dataset for the reviewer assignment problem. *ArXiv*, abs/2303.16750.

Wenbin Tang, Jie Tang, and Chenhao Tan. 2010. Expertise matching via constraint-based optimization. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 34–41.

Siyuan Wu, Leong Hou U., Sourav S. Bhowmick, and Wolfgang Gatterbauer. 2018. Pistis: A conflict of interest declaration and detection system for peer review management. In *Proceedings of the 2018 International Conference on Management of Data*, SIGMOD '18, page 1713–1716, New York, NY, USA. Association for Computing Machinery.

Yonghui Wu, Mike Schuster, Z. Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason R. Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Gregory S. Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *ArXiv*, abs/1609.08144.

David Yarowsky and Radu Florian. 1999. Taking the load off the conference chairs-towards a digital paper-routing assistant. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

Dong Zhang, Shu Zhao, Zhen Duan, Jie Chen, Yanping Zhang, and Jie Tang. 2020. A multi-label classification method using a hierarchical and transparent representation for paper-reviewer recommendation. *ACM Trans. Inf. Syst.*, 38(1).

Appendix A Paper categories used for classification-based pre-training

Field	Categories
Computer Science	Artificial Intelligence; Hardware Architecture; Computational Complexity; Computational Engineering, Finance and Science; Computational Geometry; Computation and Language; Cryptography and Security; Computer Vision and Pattern Recognition; Computers and Society; Databases; Distributed, Parallel, and Cluster Computing; Digital Libraries; Discrete Mathematics; Data Structures and Algorithms; Emerging Technologies; Formal Languages and Automata Theory; General Literature; Graphics; Computer Science and Game Theory; Human-Computer Interaction; Information Retrieval; Information Theory; Machine Learning; Logic in Computer Science; Multiagent Systems; Multimedia; Mathematical Software; Numerical Analysis; Neural and Evolutionary Computing; Networking and Internet Architecture; Other Computer Science; Operating Systems; Performance; Programming Languages; Robotics; Symbolic Computation; Sound; Software Engineering; Social and Information Networks; Systems and Control
Economics	Econometrics; General Economics; Theoretical Economics
Electrical Engineering and Systems Science	Audio and Speech Processing; Image and Video Processing; Signal Processing; Systems and Control
Mathematics	Commutative Algebra; Algebraic Geometry; Analysis of PDEs; Algebraic Topology; Classical Analysis and ODEs; Combinatorics; Category Theory; Complex Variables; Differential Geometry; Dynamical Systems; Functional Analysis; General Mathematics; General Topology; Group Theory; Geometric Topology; History and Overview; Information Theory; K-Theory and Homology; Logic; Metric Geometry; Mathematical Physics; Numerical Analysis; Number Theory; Operator Algebras; Optimization and Control; Probability; Quantum Algebra; Rings and Algebras; Representation Theory; Symplectic Geometry; Spectral Theory; Statistics Theory
Physics	Accelerator Physics; Atmospheric and Oceanic Physics; Applied Physics; Atomic and Molecular Clusters; Atomic Physics; Biological Physics; Chemical Physics; Classical Physics; Computational Physics; Data Analysis, Statistics and Probability; Physics Education; Fluid Dynamics; General Physics; Geophysics; History and Philosophy of Physics; Instrumentation and Detectors; Medical Physics; Optics; Plasma Physics; Popular Physics; Physics and Society; Space Physics; Nuclear Theory; Nuclear Experiment; Exactly Solvable and Integrable Systems; Pattern Formation and Solitons; Cellular Automata and Lattice Gases; Chaotic Dynamics; Adaptation and Self-Organizing Systems; Mathematical Physics; High Energy Physics - Theory; High Energy Physics - Phenomenology; High Energy Physics - Lattice; High Energy Physics - Experiment; General Relativity and Quantum Cosmology; Superconductivity; Strongly Correlated Electrons; Statistical Mechanics; Soft Condensed Matter; Quantum Gases; Other Condensed Matter; Materials Science; Mesoscale and Nanoscale Physics; Disordered Systems and Neural Networks; Condensed Matter; Solar and Stellar Astrophysics; Instrumentation and Methods for Astrophysics; High Energy Astrophysical Phenomena; Astrophysics of Galaxies; Earth and Planetary Astrophysics; Cosmology and Nongalactic Astrophysics; Astrophysics
Quantitative Biology	Biomolecules; Cell Behavior; Genomics; Molecular Networks; Neurons and Cognition; Other Quantitative Biology; Populations and Evolution; Quantitative Methods; Subcellular Processes; Tissues and Organs
Quantitative Finance	Computational Finance; Economics; General Finance; Mathematical Finance; Portfolio Management; Pricing of Securities; Risk Management; Statistical Finance; Trading and Market Microstructure
Statistics	Applications; Computation; Methodology; Machine Learning; Other Statistics; Statistics Theory

Table 4: 158 categories across 8 academic fields used for pre-training our paper-reviewer matching model. These categories are derived from the official categorical taxonomy of the ArXiv repository.