A Dataset for Expert Reviewer Recommendation with Large Language Models as Zero-shot Rankers

Vanja M Karan[1,5], **Stephen McQuistin**[4], **Ryo Yanagida**[3], **Colin Perkins**[3], **Gareth Tyson**[1], **Ignacio Castro**[1], **Patrick Healey**[1], **Matthew Purver**[1,2]

[1]Queen Mary University of London, [2]Jožef Stefan Institute, Ljubljana,
[3]University of Glasgow, [4]University of St Andrews, [5]University of Vienna
vanja.karan@univie.ac.at, m.purver@qmul.ac.uk

Abstract

The task of reviewer recommendation is increasingly important, with main techniques utilizing general models of text relevance. However, state of the art (SotA) systems still have relatively high error rates. Two possible reasons for this are: a lack of large datasets and the fact that large language models (LLMs) have not yet been applied. To fill these gaps, we first create a substantial new dataset, in the domain of Internet specification documents; then we introduce the use of LLMs and evaluate their performance. We find that LLMs with prompting can improve on SotA in some cases, but that they are not a cure-all: this task provides a challenging setting for prompt-based methods.

1 Introduction

Peer reviewing is critical to many processes—including scientific publication—for maintaining adequate standards (Thurner and Hanel, 2011; Bianchi and Squazzoni, 2015), and even a small numbers of poor-quality reviews can have considerable impact on the peer-review process (Robert, 1968; Triggle and Triggle, 2007; Squazzoni and Gandelli, 2012; Thurner and Hanel, 2011; Thorngate and Chowdhury, 2014; Wicherts, 2016). As the numbers of submissions and reviewers increase, manual reviewer assignment becomes infeasible (Shah, 2022). This sparks growing interest in methods to automate the reviewer assignment process (see Stelmakh et al., 2019, for an overview). While factors such as reviewer experience and conflicts of interest play a role, the core of most approaches is computing a relevance score for a reviewer-paper pair, usually via the similarity between the text of the paper and some text that characterises the expertise of the potential reviewer (e.g., previous papers authored or reviewed). The approach is therefore quite general, and can be extended to many related tasks: finding reviewers for other text types e.g., grant proposals; consultants in commercial settings; or topic experts for journalists .

However, even recent reviewer recommendation systems still have relatively high error rates (Stelmakh et al., 2023). One reason is that relevant public datasets are quite small (with just a few hundred papers/reviewers). Another may be that pretrained large language models (LLMs), which sometimes achieve good performance with little task-specific fine-tuning data on a range of tasks (Brown et al., 2020; Zhu et al., 2023), have not yet been applied to this task, and it is not clear how best to apply them.

We aim to address these gaps. First, our primary contribution is a new silver-labelled reviewer recommendation dataset, much larger than existing resources (Section 3). Second, we use retrieval-based generation and pairwise prompting to apply LLMs to the task, and evaluate this on diverse available datasets to provide a preliminary benchmark for LLMs in reviewer recommendation (Section 4). Our code and data are publicly available.[1]

2 Related work

Reviewer recommendation Factors affecting peer-review and choice of reviewers include (lack of) reviewing experience (Stelmakh et al., 2021), conflicts of interest (Resnik and Elmore, 2018), and the Matthew (*rich get richer*) effect (Robert, 1968; Squazzoni and Gandelli, 2012), but most automated paper-reviewer matching approaches rely on a measure of text similarity between a paper and a reviewer candidate, usually represented via the text of their papers. Simple but effective approaches can be found in the Toronto Paper Matching System (Charlin and Zemel, 2013), relying on count vectors and regression models. Topic models can also be used, e.g., latent Dirichlet allocation (Blei et al., 2003) and advanced variants e.g., Mimno and McCallum (2007)'s Author - Persona - Topic model or Anjum et al. (2019)'s Common Topic

[1]https://github.com/sodestream/revrec

Proceedings of the 31st International Conference on Computational Linguistics, pages 11422–11427
January 19–24, 2025.

Model. Some approaches use non-text features to characterise papers/reviewers: Tran et al. (2017) use citation networks, and Rodriguez and Bollen (2008) co-authorship graphs. Some combine multiple sources of information and/or consider multiple recommendation goals (e.g., reviewer authority). However, the appropriateness of these non-text features depends on the domain; here, our interest is in general relevance as captured from text.

Neural models Recently, results have been improved by embedding the text using pretrained neural language models, e.g., ELMo (Peters et al., 2018)'s generic embeddings; SPECTER (Cohan et al., 2020), designed for scientific documents and exploiting the citation graph; and the Multifacet-Recommender (MFR) approach of Chang and McCallum (2021), which improves SPECTER by constructing separate embeddings for different facets of the paper (and is used by the OpenReview platform). Stelmakh et al. (2023) compare these and show them to be strong baselines, but still with high error rates, from 12%-30% for easy cases to 36%-43% for hard cases. However, we are not aware of attempts to apply more recent large language models (LLMs) to this problem, although they achieve good results with limited data in related tasks (Hou et al., 2023; Sun et al., 2023; Qin et al., 2023).

Existing Datasets Here, we compare against the standard benchmark NIPS dataset (Mimno and McCallum, 2007), with paper/reviewer pairs labeled using binary relevance labels; and Stelmakh et al. (2023)'s dataset with expertise labels for paper/reviewer pairs on a 1-5 scale. A third publicly available dataset (Karimzadehgan et al., 2008), is not directly usable: it provides manually labeled topic vectors for each paper/reviewer, but no gold-standard relevance or expertise labels. Other datasets exist (e.g. Rodriguez and Bollen, 2008; Anjum et al., 2019), but not publicly available.

3 Dataset construction

To develop a large-scale dataset that can be publicly available we use data provided by the Internet Engineering Task Force (IETF).

Overview of IETF The IETF is a decentralized organization that develops the technical standards, known as *RFCs* (Request for Comments), that underpin the Internet. These standards are developed by IETF working groups (WGs) in the form of documents called *Internet-drafts*.[2] These are formally reviewed several times prior to publication as an RFC, and subject to extensive informal review discussion on mailing lists. This process is open and well documented (Bradner, 1996), and the documents, mailing list archives, and metadata about the participants is publicly available[3] and accessible via a REST API.[4] The dataset we develop contains Internet-drafts, candidate reviewers, and corresponding silver relevance labels, along with corresponding email discussion text and participant metadata. It is intended to complement the other two datasets by providing a larger freely available dataset from a different domain, including both abstracts and full documents.

We consider 3,075 Internet-drafts, comprising the final pre-publication drafts for RFCs published by the IETF between January 2010 and April 2022 inclusive. To avoid manually labeling all reviewers for relevance, we use heuristics based on the available metadata to generate three tiers of reviewer candidates, T_1, T_2, T_3, for each draft, D, in which all candidates in T_N should have higher expertise relating to D than all candidates in T_{N+1}:

- T_1 candidates are the authors of draft D and the chairs of the WG developing D. The authors are guaranteed to have very high expertise in D's area; they would obviously not be potential reviewers in practice, but are useful as training/evaluation examples of high-expertise candidates. WG chairs are selected based on a combination of technical expertise and management skills, so they can be expected to be familiar with the technical content of drafts from their WG.
- T_2 candidates are participants in the WG developing D who, during D's *discussion period*: (1) sent no messages in mailing list threads *about* D; (2) were *active* in at least one thread *not about* D; and (3) were not IETF Area Directors[5]. We define the *discussion period* as the period from D's first submission time S_1 to final submission S_2; *active* partici-

[2] An RFCis a term for the final state of an Internet draft once it is accepted and published.

[3] https://www.ietf.org/about/open-records/

[4] https://datatracker.ietf.org/

[5] WGs are organized into *areas* based on topic. Area Directors manage WG chairs and are selected for their management skills and technical breadth but are not necessarily subject matter experts in each WG they oversee; we therefore do not consider them for T_1 but also prefer to exclude them from T_2

Dataset	#papers	#reviewers	#paper-reviewer pairs
NIPS	148	364	650
Stelmakh	463	58	477
IETF	3075	1846	17562

Table 1: Dataset statistics

Dataset	abstracts	full-text	emails	participant metadata
NIPS	✓	-	-	-
Stelmakh	✓	✓	-	-
IETF	✓	✓	✓	✓

Table 2: Data availability

pants in a discussion thread as the top 20% by message-count; and that a thread is *about* D if it has the title of D in its subject.

- T_3 candidates are selected from a randomly chosen WG that is in the same area, but is not the WG that worked on D. Other conditions are identical to T_2.

The goal is that the three tiers have decreasing expertise when it comes to reviewing draft D. The expertise of T_3 is the least, as participants only in a WG that was not engaged with D; T_2 were not engaged with D but are from its WG (thus more topically related to D); and T_1, as authors and WG chairs, are experts in D's area.

Heuristics details If a participant qualifies for both tier N and $N+1$ they are assigned to tier N. T_2 and T_3 are sometimes much bigger than T_1; if so, we subsample them to be the same size in T_1. The heuristics say nothing about the relationship of candidates within the same tier; their purpose is to provide between-tier pairs of candidates for training and evaluating models.

Validation We validated these heuristics against expert judgements from two IETF participants familiar with the draft areas. We asked them to rate a randomly sampled subset of between-tier pairs as correct or not. This included 19 drafts and provided ratings for 191 pairs of candidates; the experts agreed on 183; in total only 7 were rated as wrong by at least one expert (i.e. 96% agreement).

IETF Dataset Summary. A comparison with other datasets is given in Table 1 and available data in Table 2. The IETF dataset is now the largest dataset for this task by an order of magnitude.

4 Experiments

4.1 Experimental setup

We frame the task as a retrieval problem where the set of queries $Q = \{q_1, ..., q_n\}$ are representations of documents to be reviewed, and the set of targets $T = \{t_1, ..., t_m\}$ are representations of reviewer candidates.[6] The task of the model is, given an element of Q, to order elements of T appropriately. As the paper representation q_i we use the string concatenation of its title and abstract; a reviewer representation is the string concatenation of all the paper representations authored by that reviewer.

In addition to Q and T we also have a set of annotated pairwise relevance judgements $P = \{p(q_k, t_i, t_j)\}$. Where p represents an indicator function that has the value 1 if t_i is more relevant w.r.t. q_k than t_j, and 0 otherwise. There are two ways to obtain p. First, it can be labelled directly by assigning a value to each triple depending on the correct ordering of t_i and t_j. Second, if numerical relevance scores $r(q_k, t_i)$ and $r(q_k, t_j)$ are available the value can be inferred by comparing them. We also define error weights $w(q_k, t_i, t_j)$ which describe the severity of ordering t_i and t_j incorrectly. If numerical relevance scores are available we set $w(q_k, t_i, t_j) = |r(q_k, t_i) - r(q_k, t_j)|$; otherwise if p was labelled directly we set w to 1 for all triples. Furthermore, the labels for information retrieval datasets are almost always *incomplete*, as only a fraction of all possible documents is annotated for each query. To account for this case, it is also allowed for the value of r and the corresponding w to be unknown for some triples.

To evaluate all models we use the metric from (Stelmakh et al., 2023) which considers all triples (q_k, t_i, t_j) for which the corresponding p and w are known. If the model put t_i and t_j in the wrong order this incurs a penalty of $w(q_k, t_i, t_j)$; a tie incurs half that penalty. The error is summed and normalized by the error of the worst possible model (i.e. ordering all pairs incorrectly gives an error score of 1). Intuitively this gives the percentage of annotated pairs that a model orders incorrectly, but weighted such that mistakes on target pairs with high difference in gold scores have more influence on the final score. For significance testing we use bootstrap resampling (Efron and Tibshirani, 1994) on the set of test queries. As baseline comparisons we follow

[6] The Stelmakh dataset frames Q as the reviewers and T as the documents, but our methodology is easily adapted to this.

Stelmakh et al. (2023) and use our own implementation of TPMS. We report the SPECTER+MFR results from (Stelmakh et al., 2023).

The IETF dataset has a prohibitively large number of candidate pairs. We experiment on a sample of 4,000 triplets (similar size as the other datasets).

4.2 LLM methodology

To test the potential advantage of LLMs over previous methods, we choose a single representative last-generation LLM, rather than exhaustively testing options. We selected LLaMA (Touvron et al., 2023) due to its open-source nature, relatively low resource requirements, and competitive performance compared to larger commercial alternatives.

There are three main approaches to ranking - pointwise, pairwise, and listwise (Xia et al., 2008). Both pointwise and listwise approaches have been shown to be challenging for LLMs (Qin et al., 2023). Therefore we adopt the pairwise approach.[7]

Retrieval augmented generation (RAG) Most query/target representations are too large to fit into a single prompt; however, much of the text is irrelevant for rating expertise. We therefore take the RAG approach (Lewis et al., 2020) to retrieve the most relevant parts of the text. We split q_k and t_i into sentences and compare them using embeddings derived by SentenceTransformers (Reimers and Gurevych, 2019). Each sentence in $t_{i/j}$ is scored by its average similarity to all sentences in q_k, and we take the top N (here, N=10) sentences to form the prompt $t'_{i/j}$; q_k is included as is.

Prompting Here we aim to convert a triple q_k, t_i, t_j into a preference for t_i or t_j. We therefore prompt LLaMA-2 to solicit this information. The prompt is *"[INST] Description of the paper to review is [q_k]. Description of candidate A is [t_i]. Description of candidate B is [t_j]. Which candidate is more relevant to review this paper (your answer must be "Candidate A" or "Candidate B")? [/INST] My answer is: "*[8] The model also receives an additional system prompt: *"You are an expert pairing reviewers with suitable papers to review."* If "Candidate A" or "Candidate B" appears in the model response we consider that was the better candidate, in very rare cases where this fails we consider the model decision was a tie. In this way an overall ordering between targets is established.

	Stelmakh	NIPS	IETF
TPMS	.27	.29	**.23**
TPMS (prompt)	.29	**.26**	.25
Specter+MFR	.24	-	-
LLaMA2-7b	.39*	.41*	.47*
LLaMA2-70b	**.21***	.34*	.31*
LLaMA3-8b	.34*	.35*	.30*
LLaMA3-70b	.23*	.28	.24

Table 3: Model performance (see 4.1, lower is better). * marks statistically significant difference wrt. TPMS.

4.3 Results

Results are given in Table 3. Performance on the IETF dataset are comparable to the other datasets: despite its labels being generated heuristically, it is still adequately challenging for ranking models. Expectedly, across all datasets, the larger LLaMA variants outperform the smaller ones; and LLaMA3 variants show considerable improvement over LLaMA2.

LLaMA2-70b achieves a new state of the art result (0.21) on the Stelmakh dataset, outperforming the Specter+MFR (Cohan et al., 2020) result (0.24) obtained by Stelmakh et al. (2023). However, all our LLMs fail to beat the TPMS baseline on the other two. Interestingly, LLM performance is better when the task is choosing between two papers given a reviewer (Stelmakh), than when choosing between two reviewers given a paper (NIPS, IETF). This result is consistent across LLMs and indicates the latter setting is more challenging.

We further investigate whether this issue is caused by prompts having inadequate information or by the LLM underperforming. Denoted as 'TPMS (prompt)' in Table 3, we apply the TPMS approach but with the same text representations $t'_{i/j}$; q_k as used in the LLM prompts. This gives TPMS less text information, and thus had a small detrimental effect for the Stelmakh and IETF datasets; but improved scores for NIPS, and outperforms LLMs in all settings but one, indicating that the set of prompts includes the relevant information but the LLMs are unable to fully exploit it in our completely zero-shot setting.

5 Conclusion

We introduced a new large-scale dataset with high quality silver labels for reviewer recommendation, and explored ways to apply large language models

[7] Our dataset could, in principle, be used for either of the three approaches (e.g., by assigning relevance scores of 3,2, and 1 to reviewers from T1,T2, and T3, respectively).

[8] [INST] tags separate user and model utterances.

(LLMs) in a zero-shot setting. While LLMs beat the state of the art on one dataset, the others remain a challenge; we hope this will encourage further research into applying LLMs to this task.

Future work could look into approaches that would utilize the available full document texts and other available data, especially long-context LLMs. Another avenue for improvement, which is enabled by the size of the dataset, would be to apply LLMs to this task in a few-shot or fully supervised setting. An interesting development in this vein would be investigating domain transfer between the IETF data and the standard paper peer-review domain.

6 Limitations

One limitation of the work is that we used only one family of language models. This seems sufficient for our purpose of demonstrating the challenge associated with the task and our new dataset; we have no doubt that further performance improvements could be gained by investigating other models and ways of applying them, and we hope that this will follow in future work, by ourselves or others. Another limitation is that the subset of drafts that we test the heuristic on may be biased towards the expertise of the annotators, even though we tried to include drafts from various IETF areas. Finally, our results are heavily dependent on the prompt used. While we did not have the computational resources to experiment with a wide range of prompts, we did try out several (3) variants in preliminary experiments, and selected the best one to use in all subsequent model runs.

7 Ethical Considerations

The IETF is an open standards body. Participation is dependent on accepting policies[9] that explicitly state that data about the standards process will be made public. Our analysis uses only this publicly available data. We discussed our work with IETF leadership to confirm it ensure with their policies. Our work is reproducible and we will release the code and data publicly upon acceptance.

8 Model training budget

For all experiments we used a pair of Quadro RTX 6000 GPUs with 24GB of video RAM each. A run over around 4000 candidate pairs with the largest (70b) models requires around 23 hours to complete. We used oLLaMA (https://oLLaMA.com/) to run the LLMs.

[9] https://www.ietf.org/about/note-well/

Acknowledgements

This work was partially supported by the UK EPSRC via the projects Sodestream (EP/S033564/1, EP/S036075/1), AP4L (EP/W032473/1), ARCID-UCA (EP/W001632/1) and AdSoLve (Responsible AI UK, EP/Y009800/1, project KP0016); and by the Slovenian Research Agency via research core funding for the programme Knowledge Technologies (P2-0103) and the project EMMA (L2-50070).

References

Omer Anjum, Hongyu Gong, Suma Bhat, Wen-Mei Hwu, and Jinjun Xiong. 2019. PaRe: A paper-reviewer matching approach using a common topic space. In *Proc. Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 518–528.

Federico Bianchi and Flaminio Squazzoni. 2015. Is three better than one? simulating the effect of reviewer selection and behavior on the quality and efficiency of peer review. In *2015 Winter simulation conference (WSC)*, pages 4081–4089. IEEE.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent Dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Scott O. Bradner. 1996. The Internet Standards Process – Revision 3. RFC 2026.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Haw-Shiuan Chang and Andrew McCallum. 2021. Cold-start paper recommendation using multi-facet embedding. In *Overleaf Preprint [Last accessed: Oct 2, 2023]*.

Laurent Charlin and Richard Zemel. 2013. The Toronto paper matching system: an automated paper-reviewer assignment system. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *ACL*.

Bradley Efron and Robert J Tibshirani. 1994. *An introduction to the bootstrap*. CRC press.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2023. Large language models are zero-shot rankers for recommender systems. *arXiv preprint arXiv:2305.08845*.

Maryam Karimzadehgan, ChengXiang Zhai, and Geneva Belford. 2008. Multi-aspect expertise matching for review assignment. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1113–1122.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2227–2237, New Orleans, Louisiana.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, et al. 2023. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

David B Resnik and Susan A Elmore. 2018. Conflict of interest in journal peer review.

Merton Robert. 1968. The matthew effect in science. *Science*, 159(3810):56–63.

Marko A Rodriguez and Johan Bollen. 2008. An algorithm to determine peer-reviewers. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 319–328.

Nihar B Shah. 2022. Challenges, experiments, and computational solutions in peer review. *Communications of the ACM*, 65(6):76–87.

Flaminio Squazzoni and Claudio Gandelli. 2012. Saint matthew strikes again: An agent-based model of peer review and the scientific community structure. *Journal of Informetrics*, 6(2):265–275.

Ivan Stelmakh, Nihar B Shah, and Aarti Singh. 2019. Peerreview4all: Fair and accurate reviewer assignment in peer review. In *Algorithmic Learning Theory*, pages 828–856. PMLR.

Ivan Stelmakh, Nihar B Shah, Aarti Singh, and Hal Daumé III. 2021. Prior and prejudice: The novice reviewers' bias against resubmissions in conference peer review. *Proceedings of the ACM Conference on Human-Computer Interaction*, 5(CSCW1):1–17.

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

Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. *arXiv preprint arXiv:2304.09542*.

Warren Thorngate and Wahida Chowdhury. 2014. By the numbers: Track record, flawed reviews, journal space, and the fate of talented authors. In *Advances in Social Simulation: Proceedings of the 9th Conference of the European Social Simulation Association*, pages 177–188. Springer.

Stefan Thurner and Rudolf Hanel. 2011. Peer-review in a world with rational scientists: Toward selection of the average. *The European Physical Journal B*, 84:707–711.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hong Diep Tran, Guillaume Cabanac, and Gilles Hubert. 2017. Expert suggestion for conference program committees. In *2017 11th International Conference on Research Challenges in Information Science (RCIS)*, pages 221–232. IEEE.

Chris R Triggle and David J Triggle. 2007. What is the future of peer review? why is there fraud in science? is plagiarism out of control? why do scientists do bad things? is it all a case of: "all that is necessary for the triumph of evil is that good men do nothing?". *Vascular health and risk management*, 3(1):39–53.

Jelte M Wicherts. 2016. Peer review quality and transparency of the peer-review process in open access and subscription journals. *PloS one*, 11(1):e0147913.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. In *Proc. International Conference on Machine Learning*, pages 1192–1199.

Yiming Zhu, Peixian Zhang, Ehsan-Ul Haq, Pan Hui, and Gareth Tyson. 2023. Can chatgpt reproduce human-generated labels? a study of social computing tasks. *arXiv preprint arXiv:2304.10145*.