NLPEER: A Unified Resource for the Computational Study of Peer Review

Nils Dycke, Ilia Kuznetsov, Iryna Gurevych
Ubiquitous Knowledge Processing Lab (UKP Lab)
Department of Computer Science and Hessian Center for AI (hessian.AI)
Technical University of Darmstadt
ukp.informatik.tu-darmstadt.de

Abstract

Peer review constitutes a core component of scholarly publishing; yet it demands substantial expertise and training, and is susceptible to errors and biases. Various applications of NLP for peer reviewing assistance aim to support reviewers in this complex process, but the lack of clearly licensed datasets and multi-domain corpora prevent the systematic study of NLP for peer review. To remedy this, we introduce NLPEER – the first ethically sourced multidomain corpus of more than 5k papers and 11k review reports from five different venues. In addition to the new datasets of paper drafts, camera-ready versions and peer reviews from the NLP community, we establish a unified data representation and augment previous peer review datasets to include parsed and structured paper representations, rich metadata and versioning information. We complement our resource with implementations and analysis of three reviewing assistance tasks, including a novel guided skimming task. Our work paves the path towards systematic, multi-faceted, evidence-based study of peer review in NLP and beyond. The data[1] and code[2] are publicly available.

1 Introduction

Research publication is the primary unit of scientific communication. To ensure publication quality and to prioritise research outputs, most scientific communities rely on peer review (Johnson et al., 2018) – a distributed procedure where independent referees determine if a manuscript adheres to the standards of the field. Despite its utility and wide application, peer review is an effortful activity that requires expertise and is prone to bias (Tomkins et al., 2017; Lee et al., 2013; Stelmakh et al., 2021). An active line of research in NLP for peer review strive to address these challenges by supporting the underlying editorial process (e.g. Price and Flach, 2017; Kang et al., 2018; Shah, 2019), decision making (e.g. Shen et al., 2022; Dycke et al., 2021; Ghosal et al., 2019), review writing (Yuan et al., 2022), and by studying review discourse (e.g. Kennard et al., 2022; Kuznetsov et al., 2022; Hua et al., 2019; Cheng et al., 2020; Ghosal et al., 2022b).

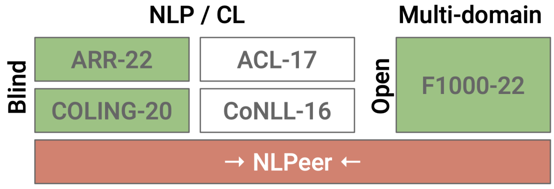

Figure 1: NLPEER unites openly licensed datasets from different research communities, reviewing systems and time periods, including three previously unreleased text collections: ARR-22, COLING-20 and F1000-22.

Despite the methodological advancements, several factors prevent NLP research for peer review at large. The computational study of peer review lacks a (1) **solid data foundation**: reviewing data is rarely public and comes with legal and ethical challenges; existing sources of peer reviewing data and the derivative datasets are not licensed, which legally prevents reuse and redistribution (Dycke et al., 2022). Peer reviewing practices vary across research communities (Walker and Rocha da Silva, 2015; Bornmann, 2011) – yet the vast majority of NLP research in peer review so far focused on a few machine learning conferences that make their data available through the OpenReview.net platform (e.g. Kennard et al., 2022; Shen et al., 2022). A (2) **multi-domain perspective on peer review** is thus missing, and the transferability of findings between different communities and reviewing workflows remains unclear. Finally, a (3) **unified data model** for representing peer reviewing data is lacking: most existing datasets of peer reviews adhere to task-specific data models and formats, making it hard to develop and evaluate approaches for peer

[1] https://tudatalib.ulb.tu-darmstadt.de/handle/tudatalib/3618
[2] https://github.com/UKPLab/nlpeer

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5049–5073
July 9-14, 2023 ©2023 Association for Computational Linguistics

review support across datasets and domains.

To address these issues, we introduce NLPEER. We apply a state-of-the-art workflow (Dycke et al., 2022) to gather ethically and legally compliant reviewing data from natural language processing (NLP) and computational linguistics (CL) communities. We complement it with multi-domain reviewing data from the F1000 Research[3] platform and historical data from the openly licensed portion of the PeerRead (Kang et al., 2018) corpus. The resulting resource (Figure 1) is the most comprehensive collection of clearly licensed, open peer reviewing datasets available to NLP to date.

NLPEER includes peer reviews, paper drafts and revisions from diverse research fields and reviewing systems, over the time span from as early as 2012 until 2022. This – for the first time – enables systematic computational study of peer review across domains, communities, reviewing systems and time. The paper revisions make NLPEER well-suited for the study of collaborative text work.

To facilitate the analysis, we unify the datasets under a common data model that preserves document structure, non-textual elements and is well suited for cross-document analysis. To explore the new possibilities opened by our resource, we conduct cross-domain experiments on *review score prediction* (to encourage consistent review scores), *pragmatic labeling* (to encourage balanced reviews) and the novel *guided skimming for peer review* task (to help guide review focus), along with easy-to-extend implementations. Our results indicate substantial variation in performance of NLP assistance between venues and research communities, point at synergies between different approaches to review structure analysis, and pave the path towards exploiting cross-document links between peer reviews and research papers for language model benchmarking.

In summary, this work contributes (1) the first unified, openly licensed, multi-domain collection of datasets for the computational study of peer review, including (2) two novel datasets of peer reviews from the NLP and CL communities, and complemented by a (3) descriptive analysis of the resulting data and (4) extensive experiments in three applied NLP tasks for peer reviewing assistance.

2 Background

2.1 Peer Reviewing Terminology

During peer review, authors submit their *paper*[4] to the editors, often via a peer reviewing platform. The manuscript is distributed among reviewers who produce reviewing reports – or *reviews* – evaluating the submission. As a result, the submission might be accepted, rejected or further adjusted, producing a *revision*. Some reviewing systems allow additional exchange, e.g. author responses and meta-reviews. However, here we focus on papers, reviews and revisions, which constitute the core of *peer reviewing data*.

There exist different implementations of peer review, including *blind review* (where reviewer and/or author identities are hidden to promote objectivity) and *open review* (identities are open). Moreover, reviewing standards and practices vary by research community, venue and publication type. Review forms and templates are one important varying factor; in the following, we differentiate between *unstructured* (only one main text) and *structured* (several predefined sections) review forms. We refer to the particular implementation of peer review at a certain venue as *reviewing system*. Along with the natural domain shift based on research field, the reviewing system contributes to the composition of the peer reviewing data it produces.

2.2 Existing Data

Peer review is a hard, time-consuming, subjective task prone to bias, and an active line of work in NLP aims to mitigate these issues. NLP for peer review crucially depends on the availability of peer reviewing data – yet open data is scarce. The majority of existing NLP studies on peer review (e.g. Kang et al., 2018; Yuan et al., 2022; Hua et al., 2019; Kennard et al., 2022; Cheng et al., 2020; Ghosal et al., 2022a; Shen et al., 2022) draw their data from a few machine learning conferences on the OpenReview.net platform which – at the time of writing – does not attach explicit licenses to the publicly available materials, making reuse of the derivative data problematic. In addition, over-focusing on few selected conferences in machine learning limits the utility of NLP approaches to peer review for other areas of science and reviewing systems, and leaves the question of cross-domain applicability open. While the recent F1000RD corpus

[3] https://f1000research.com/

[4] We use the terms manuscript, paper etc. interchangeably.

(Kuznetsov et al., 2022) addresses some of these issues, it lacks data from NLP and CS communities, and does not contain blind reviewing data, although single- and double-blind review are arguably standard in most research fields (Johnson et al., 2018).

Ethical, copyright- and confidentiality-aware collection of peer reviewing data (Dycke et al., 2022), and transformation of this data into unified, research-ready datasets, require major effort. The purpose of NLPEER is to provide the NLP community a head-start in ethically sound, cross-domain and cross-temporal study of peer review.

2.3 NLP Approaches to Aiding Peer Review

Recent years have seen a surge in NLP approaches to aid peer review; ranging from early works studying reviewing scores (Kang et al., 2018) to more recent approaches attempting to align author responses to review reports automatically (Kennard et al., 2022). The goal of our work is to provide diverse and clearly licensed source data to support the future studies in NLP for peer review assistance.

In addition, we explore the potential for cross-domain NLP-based peer reviewing assistance on three tasks detailed in Section 5. Review score prediction has been first introduced in (Kang et al., 2018) and further explored in (Ghosal et al., 2019). While following a similar setting, our study contributes to this line of research by exploring the task of score prediction across domains and research communities and provides new insights on the factors that impact the transferability of the task between reviewing systems. Pragmatic labeling has been previously explored in (Hua et al., 2019; Kuznetsov et al., 2022; Kennard et al., 2022) and is usually cast as a discourse labeling task on free-form peer review text. Yet, many reviewing systems enforce the same discourse structure by employing structured peer review forms. Complementing the prior efforts, we explore the potential synergies between the two approaches. The guided skimming for peer review task is novel and builds upon the recent work in cross-document modeling for peer review by Kuznetsov et al. (2022) and other related works (Qin and Zhang, 2022; Ghosal et al., 2022a).

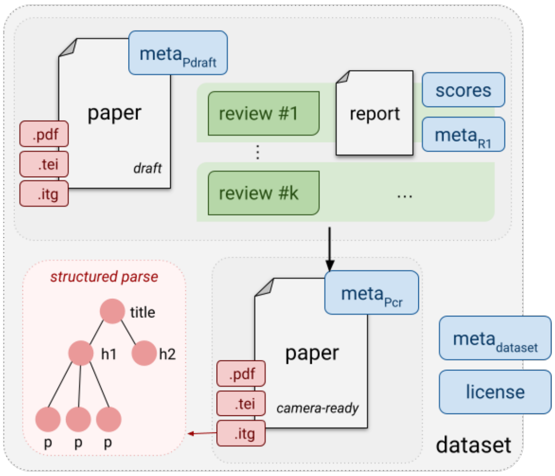

Figure 2: Unified data structure of NLPEER.

3 NLPEER

3.1 Datasets

NLPEER consists of five datasets: two datasets from previous work, two entirely new datasets from the NLP domain, as well as an up-to-date crawl of the F1000Research platform.

Prior datasets Parts of the PeerRead data (Kang et al., 2018) have been created with explicit consent for publication and processing by both reviewers and authors, and we include them into our resource. **ACL-17**[5] and **CONLL-16**[6] contain peer reviews and papers from the NLP domain, stem from a double-blind reviewing process, and use unstructured review forms with a range of numerical scores, e.g. substance and soundness. The data is licensed under CC-BY.

F1000-22 is collected from F1000Research – a publishing platform with an open post-publication reviewing workflow. Unlike other datasets in NLPEER, F1000-22 covers a wide range of research communities from scientific policy research to medicine and public health. The reviewing process at F1000Research is fully open, with reviewer and author identities known throughout the process, contributing to the diversity of NLPEER. F1000Research uses unstructured peer reviewing forms coupled with a single 3-point overall score (approve, reject, approve-with-reservations). The paper and review data are distributed under CC-BY license, which we preserve.

[5] 55th Annual Meeting of the Association for Computational Linguistics (ACL)

[6] Conference on Computational Language Learning 2016

	New Datasets		Existing Datasets			
	ARR-22	**COLING-20**	**ACL-17**	**CONLL-16**	**F1000-22**	**total**
reviewing system	blind	blind	blind	blind	open	both
domain	NLP/CL	NLP/CL	NLP/CL	NLP/CL	multi	multi
time (20XX)	21-22	20	16-17	15-16	12-22	12-22
# papers	476	89	136	22	4949	5672
% accepted	100%	93%	67%	50%	34%	69%*
# reviews	684	112	272	39	10418	11515
reviews per paper	1.43 ± 1.1	1.27 ± 0.5	2.0 ± 0.8	1.77 ± 0.6	2.11 ± 0.71	1.72*
#sent. per paper	220 ± 62.9	195 ± 47.9	208 ± 42.3	199 ± 35.6	158 ± 90.3	196*
#sent. per review	31 ± 27.0	30 ± 22.8	41 ± 24.6	32 ± 12.4	34 ± 27.6	32.2*
total #tok reviews	266k	45k	100k	16k	3.8M	4.2M

Table 1: For the datasets in NLPEER, we report the mean and standard deviation of per-paper and per-review statistics. We report % of accepted papers; for F1000-22 this is the % of version one drafts with unanimous scores of "accept", as acceptance conceptually does not exist here. Total statistics are summed and averaged (*), respectively.

COLING-20 (New) was collected via a donation-based workflow at the 28th International Conference on Computational Linguistics, in the NLP and computational linguistics domain. The data stems from a double-blind reviewing process; review forms include free-form report texts and multiple numerical scores, e.g. relevance and substance. We release this data under the CC-BY-NC-SA 4.0 license.

ARR-22 (New) was collected via the donation-based workflow proposed by Dycke et al. (2022) at ACL Rolling Review (ARR) – a centralized system of the Association for Computational Linguistics. NLPEER includes peer reviewing data for papers later accepted at two major NLP venues – ACL 2022[7] and the NAACL 2022[8] – covering submissions to ARR from September 2021 to January 2022. The reviewing process at ARR is double-blind and uses standartized structured review forms that include strengths and weakness sections, overall and reproducibility scores, etc. We release this data under the CC-BY-NC-SA 4.0 license.

3.2 Unification

The diverse source datasets in NLPEER were cast into a unified data model (Figure 2). Each paper is represented by the submission version and a revised camera-ready version, and is associated with one or more review reports. In the case of F1000-22 all revisions and their reviews are present. To unify the papers, we converted all drafts and revisions in NLPEER into intertextual graphs (ITG) – a recently proposed general document representation that preserves document structure, cross-document links and layout information (Kuznetsov et al., 2022). We extended the existing ITG parser[9] and combined it with GROBID (GRO, 2008–2023) to process PDF documents and preserve line number information whenever available. Papers were supplemented with the PDF, XML and TEI source whenever available. Reviews were converted into a standardized format that accommodates structured and free-text reviews and arbitrary sets of scores. As paper revisions were not collected for some of the NLPEER datasets, we have complemented existing data with camera-ready versions obtained via the ACL Anthology[10]. Papers, reviews and datasets are accompanied with metadata, e.g. paper track information and licenses for individual dataset items. Further dataset creation details are provided in the Appendix A.

3.3 Ethics, Licensing and Personal Data

All datasets included in NLPEER are distributed under an open Creative Commons license and were collected based on explicit consent, or an open license attached to the source data. The ARR-22 data collection process allowed reviewers to explicitly request attribution, and this information is included in the dataset. F1000Research uses open

[7] 60th Annual Meeting of the ACL

[8] 2022 Annual Meeting of the NAACL

[9] https://github.com/UKPLab/intertext-graph

[10] https://aclanthology.org/

reviewer and author identities throughout the reviewing process; this information is preserved in F1000-22. Finally, the authors of camera-ready publications included in NLPEER are attributed. In all other cases, review reports and paper drafts throughout NLPEER are stripped of personal metadata; in addition, the reviews of all datasets apart from F1000-22 have been manually verified by at least one expert from our team to not contain personal information. Due to the practices of scientific publishing in the selected communities, NLPEER only includes texts written in English.

4 Statistics

The datasets in NLPEER originate from different domains and peer reviewing systems, each with its own policies, reviewing guidelines, and community norms. While a comprehensive comparison of publishing and reviewing practices lies beyond the scope of this work, this section provides a brief overview of the key statistics for NLPEER datasets.

Documents and Texts Table 1 reports the textual statistics of NLPEER, comprising more than 4M peer review tokens in total, with more than 400K review tokens in the NLP/CL domain. The resource is diverse, and we observe high variability in review and paper composition among the datasets. For example, papers in F1000-22 come from a wide range of domains and cover a wide range of article types from case reports to position papers, reflected in the low number of sentences per paper (158) with high variance(±90.3) compared to the other, NLP-based datasets with roughly 200 sentences per paper and lower variance. Yet, we note that review lengths exhibit smaller variance across the datasets. Every paper in NLPEER is associated with at least one review and one revision, making it suitable for the study of cross-document relationships.

Scores and Acceptance The data collection workflow impacts the proportion of accepted papers in the dataset: from 100% in ARR-22 which uses a strict confidentiality-aware collection procedure, to 34% in F1000-22, where manuscripts are made available prior to peer reviewing and acceptance. Yet, acceptance per se is not an accurate proxy of review stance: a paper that is eventually accepted can receive critical reviews. To investigate, we turn to reviewing scores. Each of the reviewing systems in NLPEER requires reviewers to assign a numerical rating to the papers. However, scoring scales and semantics differ across datasets: the NLP and CL conferences employ fine-grained scales (5- and 9-point), while F1000 Research uses a very coarse scale (3-point). Figure 3 shows the normalized distribution of overall scores for each dataset in NLPEER. We observe that scores near the arguably most interesting region around the borderline to acceptance are well-represented, with a skew towards the positive end of the review score scale otherwise. For the donation-based datasets (all except F1000-22), this is likely due to participation bias (Dycke et al., 2022); for F1000-22, it is a result of the post-publication, revision-oriented reviewing workflow. This fundamental difference between the peer reviewing systems in NLPEER makes it an interesting and challenging target for the computational study of reviewing scores across reviewing systems and research communities.

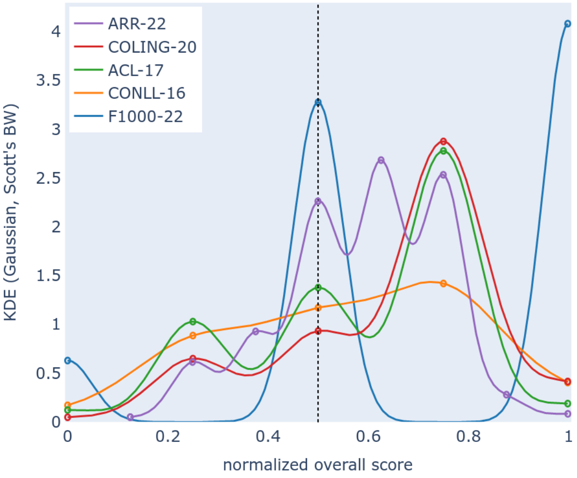

Figure 3: Normalized overall score per dataset (KDE estimate) with "borderline" (dotted).

Domain Structure The vocabulary of a text collection is an important proxy for describing its language variety (Plank, 2016), and higher vocabulary overlap indicates shared terminology and topical alignment between datasets. To investigate the domain structure of NLPEER, we measure the vocabulary overlap of review texts based on the Jaccard metric for the top 10% most frequent lemmas excluding stopwords similar to Zhang et al. (2021). As Figure 4 demonstrates, reviews from the NLP/CL communities (ARR-22, ACL-17, COLING-20) are most similar (0.37-0.53), while F1000-22 is most similar to ARR-22 with a notably lower score than the within-community comparison ($\Delta \approx 0.25$). This illustrates that despite domain differences, the review reports do share general

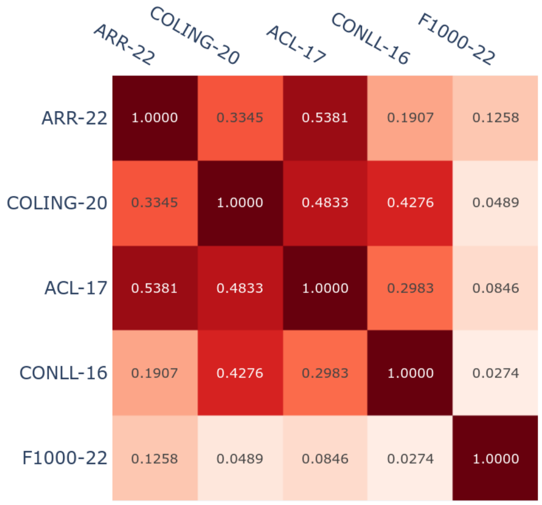

Figure 4: Vocabulary overlap based on the Jaccard metric on review lemmas of the datasets in NLPEER.

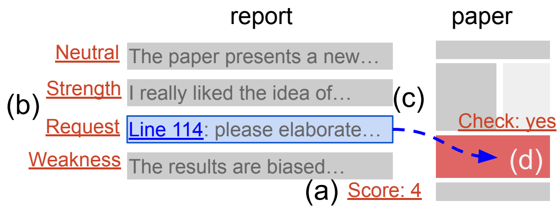

Figure 5: NLPEER tasks: given a review and a paper, we (a) predict the review score, (b) predict pragmatic class per review sentence, and (c) use explicit links to (d) find review-focus paragraphs in new papers.

wording (e.g. "model", "method", "author") that is independent from lower-frequency field-specific terminology (e.g. "cross-entropy", "feed-forward", "morpheme"). NLPEER includes datasets with linguistically diverse review reports while maintaining a domain-independent base vocabulary characteristic for the genre of peer reviews.[11] This makes the investigation of cross-domain NLP for reviewing assistance a promising research avenue.

5 Reviewing Assistance with NLPEER

NLPEER is a unique resource for the study of computational assistance during peer review authoring. To demonstrate its versatility, we define three tasks – review score prediction, pragmatic labeling and guided skimming – that are anchored in realistic assistance scenarios and targeted towards helping junior reviewers to improve their review. Figure 5 illustrates the tasks. For readability, we use s to denote sentences, *par* for paragraphs, and *sec* to denote sections for reviews (e.g. s^R) and papers (e.g. s^P), respectively. We report the main results here, and refer to the Appendix B for details and the Limitations Section for an ethical discussion of risks and opportunities of these tasks.

5.1 General Setup

Baseline Models Our experiments aim to assess the difficulty of each assistance task across the subsets of NLPEER. Hence, we base our experiments on well-established large language models (LLM) from the general and scientific domain – RoBERTa (Liu et al., 2019), BioBERT (Lee et al., 2020) and SciBERT (Beltagy et al., 2019) – with the recommended default hyperparameter settings.

Fine-tuning and Evaluation For each task, we split the data into a training (70%), development (10%) and test set (20%), and fine-tune the pretrained LLM on the training split using a task-specific prediction head. To account for small dataset sizes, we fine-tune each model with small learning rates ($[1, 3 \times 10^{-5}]$) and a linear warm-up schedule, and allow training for up to 20 epochs, following existing recommendations (Mosbach et al., 2021). We repeat the experiments ten-times with different random seeds and report the median and standard deviation of each performance metric across the runs. All experiments were run on a single NVIDIA A100 GPU, summing to a total of roughly seven days computing time all together.

5.2 Review Score Prediction

Reviewer rating behavior is heterogeneous; the correspondence between review ratings and review texts often lacks consistency (Wang and Shah, 2019). The differences in rating assignment might exist both on the individual level (i.e. due to lack of experience) and on the community level (i.e. when reviewing for a large multi-track conference). Here, a review score prediction model can suggest a score to the reviewer that is typical for the given report and sub-community. This suggestion can serve as feedback to revise inconsistent scores.

Task. Following prior work (Kang et al., 2018; Ghosal et al., 2019; Stappen et al., 2020), we cast review score prediction (RSP) as a regression task where given the review text in R and the paper abstract a^P the model should predict the overall review score q^R mapped to the respective scale. We provide the paper abstract as an input to the

[11] The paper vocabulary overlap follows a similar trend, see Appendix (Figure 7).

model to allow to contextualize the review text; hereby, the model can resolve coreferences and weigh review statements in relation to the paper. We measure the performance by the mean root squared error (MRSE). As one key challenge of review scoring is the mapping of an overall assessment to a discrete score scale, we also measure the classification performance of the regression model in terms of the F1-score by splitting the real-valued outputs into equally sized intervals according to the respective review score scale. We highlight that this task framing does not account for specific score semantics, but in exchange permits direct model transfer and comparison across domains. We leave the in-detail study of RSP as a classification task, for instance using in-context learning (Brown et al., 2020) on score semantic labels, to future work.

Setup. For each dataset in NLPeer, we consider all reviews of the initial paper draft; this means, for F1000-22 we include only the reviews of the first paper version. As the input, we concatenate the review report sections $sec_1^R, sec_2^R ... \in R$ and the paper abstract a^P, which we separate by a special token. To account for the limitations of the LLMs, we truncate the resulting text to 512 tokens. For training, we normalize the scores q^R to the interval $[0, 1]$ considering the maximum and minimum scores of the respective rating scale. The same paper can receive multiple reviews; we ensure that all reviews of the same paper belong to the same data split. For representative sampling, we ensure that the distribution of reviews per paper is similar across splits (see Appendix B).

Results. Table 2 summarizes score prediction results for the best-performing LLM by dataset. Neural models substantially outperform the mean score baseline for ARR-22 and F1000-22, yet for ACL-17, CONLL-16 and COLING-20 they perform on-par or worse. We note that the latter datasets also have the lowest number of samples. To investigate a potential connection, we have trained a RoBERTa model on 30% of the ARR-22 training set, resulting in a median 0.44 MRSE and 0.24 F1-macro score on the test set – on par with the mean baseline, and comparable to the similarly-sized ACL-17. This suggests that the dataset size has the strongest impact on score prediction performance, encouraging future reviewing data collection efforts, as well as the study of few-shot transfer between high- and low-resource peer reviewing systems and domains.

	MRSE ↓	**F1-macro** ↑
ARR-22†	.45 / **.37** ±.02	.24 / **.46** ±.04
COLING-20†	**.22** / .45 ± .08	**.23** / .15 ± .01
ACL-17‡	.78 / .76 ± .06	.06 / **.11** ±.04
CONLL-16†	.78 / .77 ± .04	.08 / .08 ± .00
F1000-22‡	.41 / **.21** ±.01	.19 / **.41**±.04

Table 2: RSP performance: baseline (mean-score) / median ± standard deviation. MRSE lower is better, F1 higher is better. Reporting best performing LLM; † = RoBERTa, ‡ = BioBert; full results in Appendix B.

5.3 Pragmatic Labeling

The core function of peer review is to assess and suggest improvements for the work at hand: a good review summarizes the work, lists its strengths and weaknesses, makes requests and asks questions. Although some venues employ structured review forms to encourage review writing along these dimensions, issues of imbalanced feedback (e.g. focus only on weaknesses) (Hua et al., 2019), inconsistent uses of review sections, and lack of guidance for free-form reviews persist. Here, a pragmatic labeling model can provide feedback to reviewers to potentially revise, balance or rearrange their review report. Following up on prior studies on argumentative and pragmatic labeling schemata for free-form peer reviews (Hua et al., 2019; Kuznetsov et al., 2022, etc.), we use NLPeer to explore the connection of pragmatic labels and structured review forms for the purpose of this assistance scenario.

Task We cast pragmatic labeling as a sentence classification task, where given a review sentence s_i^R, the model should predict its pragmatic label $c_i \in C$. For this experiment we use the data from two distinct reviewing systems: F1000Research employs free-form full-text reviews; a subset of F1000-22 has been manually labeled with sentence-level pragmatic labels in the F1000RD corpus (Kuznetsov et al., 2022). ACL Rolling Review, on the other hand, uses structured review forms split into sections, which we align to the F1000RD classes: `Strength`, `Weakness`, `Request` (`Todo` in F1000RD) and `Neutral` (`Recap`, `Other` and `Structure` in F1000RD). See Appendix B for more details.

Setup For F1000RD we consider all 5k labeled sentences mapped to the respective classes. For ARR-22, we take review sentences longer than five characters (to filter out splitting errors), and label

	ARR-22	F1000RD
ARR-22†	0.72 ± 0.01	0.50 ± 0.01
F1000RD†	0.54 ± 0.01	0.87 ± 0.01
baseline (maj.)	0.11	0.152

Table 3: Pragmatic labeling performance using †RoBERTa. Rows: training. Columns: test. Macro-F1, median ± standard deviation, *vs* majority class baseline.

them by their respective review section, resulting in around 14k labeled sentences. For each dataset, we split the instances at random disregarding their provenance to maximize the diversity across splits.

Results Table 3 presents the pragmatic labeling results in- and cross-dataset. Expectedly, neural models outperform the majority baseline in-dataset by a large margin. Yet, cross-dataset application also yields non-trivial results substantially above the in-dataset majority baseline, despite the domain shift between the ARR-22 and F1000RD data. This has important implications, as it suggests that data from structured reviewing forms (entered by reviewers as part of the reviewing process) can be used to train free-text pragmatic labeling models, thereby significantly reducing the annotation costs. Nevertheless, the gap to the in-distribution supervised model remains, constituting a promising target for the follow-up work, that would need to disentangle the effects of domain, task and reviewing system shift on pragmatic labeling performance.

5.4 Guided Skimming for Peer Review

Review writing typically requires multiple passes over the paper to assess its contents. Different paper types (e.g. dataset or method papers) require different reviewing strategies (Rogers and Augenstein, 2020); hence, the regions that require most scrutiny and rigor during reading vary across papers. Suggesting passages most relevant to the required reviewing style could encourage higher quality of reviewing and serve as a point of reference to junior reviewers. We model this scenario via the novel *guided skimming for peer review* task, in line with Fok et al. (2023) who integrate passage recommendations into reading environments, reporting improved reading performance.

Task We model the task as follows: given a paper, the model should rank its paragraphs par^P by relevance to the critical reading process. The training data for this task is derived from *explicit links*, e.g. mentions of line numbers or sections in the reviews, which can be reliably extracted from reports in a rule-based fashion (see B.3 for details) and used to draw cross-document links between review report sentences s^R and the paragraphs of the papers they discuss par^P (Kuznetsov et al., 2022)[12]. Paper paragraphs with incoming explicit links are then considered "review-worthy", and the task is to rank such paragraphs above others for previously unseen papers with no available review reports. While the resulting task is a simplification of the actual skimming process during peer review, it is a first step towards exploring review-paper-links for modeling actual reviewer focus. We encourage future work to follow-up on this line of inquiry.

Setup We use the ARR-22 and ACL-17 datasets, as they are of sufficient size and offer line numbers in the paper drafts. We extend and apply the explicit link extractor proposed by Kuznetsov et al. (2022), resulting in a total of 229 papers with 743 relevant passages with an average of 3.24 ± 3.2 of linked passages for ARR-22 and 87 papers, 308 passages and 3.54 ± 3.13 linked paragraphs per paper for ACL-17. We fine-tune LLMs using a binary classification objective, batching linked and unlinked paragraphs of the same paper together and rank by the output softmax of the positive score. We compare to a random baseline. Datasets are split by paper while ensuring a similar distribution of passages per paper across splits (see app. B).

Results Figure 6 summarizes the Precision and Recall at k for the best performing SciBERT model and the random baseline on ACL-17. Both recall and precision exceed the random baseline by a large margin and for all considered k. At around $k = 3$ roughly 50% of the relevant passages are retrieved; while at the same rank around 21% of the retrieved paragraphs are actually linked by the reviewers. The mean reciprocal rank (MRR) measures the average position of the first relevant result within the rankings. SciBERT achieves an MRR of 0.41 ± 0.05 on ACL-17 and 0.34 ± 0.03 on ARR-22, outperforming the random baseline by 0.23 and 0.18, respectively. Overall, the LLM perform substantially above random despite discarding all context information of a paragraph (see appendix C for a detailed analysis). While *guided skimming* is

[12] We opt against *implicit links*, i.e. links without explicit markers, as their automatic detection remains challenging.

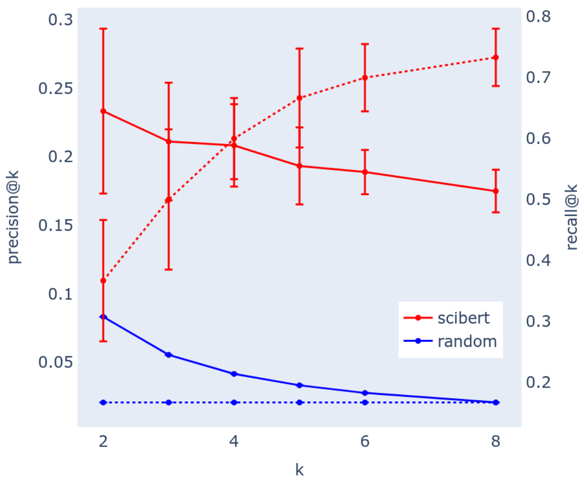

Figure 6: Precision (left) and recall (right, dotted lines) at k for SciBERT and random baseline on ACL-17.

a non-trivial task, the above-random performance of the given LLM baseline shows promise for further research in this direction, which could also include an in-depth study considering the context of paragraphs – e.g. their position in the logical structure of the paper.

6 Further Applications

NLPEER includes rich representations and metadata for a large number of inter-linked peer review and manuscript texts. This enables new NLP studies for peer reviewing assistance and beyond. In this section we highlight and critically reflect on further applications of NLPEER in general NLP research and in practice.

Peer review reports are expert-written texts that typically reflect deep understanding of the underlying paper. Hereby, they can serve as a valuable resource for distant or direct supervision of general-purpose machine learning models. For instance, paper summaries in review reports can be used as a basis for a challenging document summarization task; aspect score disagreements between reviews may serve as a weak supervision signal for aligning arguments for and against the paper across reviews.

On the other hand, NLPEER and peer reviewing data in general might be exploited to train models with dual uses and practical risks. One such example is automatic review generation, where given a paper the model should generate a review report. While a resulting model may be used benevolently as a paper-writing aid or an analytical device for the study peer review, using it to avoid the reviewing effort or replace the reviewer bears a wide range of risks. As it is implausible that current state-of-the-art NLP models could produce a non-generic, meaningful review of a novel paper (Yuan et al., 2022), such applications could compromise the academic quality assurance process. Due to the central role of peer review in research and publishing, we emphasize that future work should carefully reflect on the real-world impact of any models developed based on NLPEER.

7 Conclusion and Intended Use

We have presented NLPEER– the first clearly licensed large-scale reference corpus for the study of NLP for peer review. NLPEER opens many new opportunities for the empirical study of scholarly communication. For NLP, it allows developing new annotated datasets based on clearly licensed, richly formatted, unified corpora that span multiple research domains, reviewing systems and time periods. It can be used as a testing ground for domain transfer (Gururangan et al., 2022; Chronopoulou et al., 2022), and enables the study of cross-document relationships between papers, paper revisions and peer reviews (Kuznetsov et al., 2022). From the meta-scientific perspective, it allows comparing peer reviewing practices across research communities, and can provide crucial insights into how researchers review and revise scientific texts. Our task implementations and results can guide the development of NLP assistance systems and can be used for systematically comparing pre-trained language models in the context of peer reviewing applications. Finally, our resource can serve as a blueprint for future aggregate resources for the study of peer review in NLP.

Acknowledgements

We would like to thank all parties that supported and advised us during the realization of the peer review data collection at ACL Rolling Review. ARR-22 would not have been possible without the discussion and approval by the ACL Committee on Reviewing in 2021 chaired by Hinrich Schütze, and we are grateful to everyone who reached out to us during the pilot stages of the project to make suggestions and express their concerns. We thank the editors-in-chief and the technical team of ACL Rolling Review for their support during the data collection; with a special thanks to Amanda Stent, Sebastian Riedel and Goran Glavaš. Additionally, we would like to thank the OpenReview.net team for their support during the implementation of the data collection at ARR. We express our gratitude to Nuria Bel for her feedback during the first iterations of our data collection initiative at COLING-2020, and to Richard Gerber for helping us with early technical challenges in SoftConf. Last but not least, we would like to thank our reviewers for their valuable suggestions, as well as the community members who have engaged in a lively debate on this initiative and provided us with both encouragement and useful feedback.

This research work has been funded by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE. It is co-funded by the European Union (ERC, InterText, 101054961). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. Finally, parts of this work have been co-funded by the German Research Foundation (DFG) as part of the PEER project (grant GU 798/28-1). This work is part of the InterText initiative[13].

Limitations

While we hope that our approach to data collection can serve as a benchmark for future NLP studies beyond peer review, we deem it equally important to explicitly outline the potential risks and limitations of NLPEER and NLP for peer review in general. Our discussion below encourages future research in ethics and applied NLP for peer review; many of our considerations are not specific to peer review and are equally relevant to the applications of NLP in general.

From the data perspective, we deem it important to clearly state what NLPEER is **not meant for**. Our data collection campaigns for ARR-22 and COLING-20 included an explicit disclaimer on the risks of author profiling on the peer reviewing data; we stress that such applications **violate the intended use of NLPEER**. Furthermore, NLPEER enables a wide range of new NLP assistance tasks for peer review. Yet, we encourage future studies of NLP for peer review to reflect carefully about the **potential risks and benefits** of new task definitions atop of peer reviewing data in general and NLPEER specifically. For instance, the full automation of peer review, i.e. the generation of review reports given a paper, bears risks and dual uses.

Considering **diversity** in NLP datasets, we stress that even NLPEER only covers a fraction of peer reviewing across all fields of science, and more data needs to be collected to enable fully representative NLP-based study of peer review. Due to the genre standards of scientific publishing our dataset only covers papers and reviews in **English language**. Multilingual scholarly document processing is overall poorly represented in NLP, and constitutes a promising avenue for future research. While our resource contains data from a wide range of domains, **research in arts and humanities is under-represented** due to the poor data availability. The trend towards open science and the adoption of responsible data collection practices (Dycke et al., 2022) might bring reviewing data from previously unexplored domains and languages into NLP. We stress that any direct comparison based on our corpus would need to take into account reviewing practices and guidelines adopted by the respective communities. Specifically, potential biases resulting from the donation-based collection for ARR-22 and COLING-20 should be taken into account.

From the task side, we highlight that implementations and resulting models presented here are meant to **exemplify** the proposed tasks, determine their technical feasibility, and **serve as a starting point for developing future NLP for peer review assistance systems**. As such, the provided implementations have **limitations**: for example, sentence-level pragmatic labels derived from structure-based ARR forms might contain noise since ARR forms group text on section level; guided skimming does not make use of implicit links, and explicit links are mostly based on line numbers and quotes, limiting the recall. Since we did not perform extensive hyperparameter search and tuning of the models, our results should *not* be interpreted as a claim towards superiority of a particular model, approach or reviewing system.

We highlight that high intrinsic task performance does not necessarily translate into the extrinsic utility of NLP support in real-world reviewing environments. We thus deem it crucial to study the factors that affect the success of NLP assistance for peer reviewing. This includes the study of the human-machine interaction dynamics and its desiderata;

[13] https://intertext.ukp-lab.de/

for example, review score recommendations should be accompanied by explanations. We encourage extensive research on **risks of biases and errors** in NLP assistance models; for instance, a review score prediction model might learn undesirable biases against certain types of papers. Review writing assistance implemented in a real reviewing system should always be accompanied by carefully designed guidelines and policies.

Finally, we invite the community to reflect on the **potential societal consequences** of the individual NLP assistance tasks, *even if NLP models accomplish them well*. To provide an example, our newly introduced guided skimming task assists during the effortful and time-intensive, yet crucial step of reading the paper under review. Although the guided skimming for peer review task models an intermediate step during reading and is intended to serve as an additional point of reference during the iterated skimming steps of peer review, such a technology might encourage reviewers to read *only* the paragraphs suggested by the model. We argue that this risk of "lazy reading" is independent of the technology at hand; a reviewer that is institutionally incentivized to perform reviews as quick as possible, may read a paper superficially and settle with heuristics for their assessment (Rogers and Augenstein, 2020) regardless of assistance. A greater risk, however, may be imposed by potential biases and errors of a guided skimming model, which could distract less experienced reviewers. While recent work on skimming assistance in scholarly articles (Fok et al., 2023) suggests a mature and reflected interaction of users with highlight recommendations and possible errors, this needs a specific investigation for the use case during peer review. On the other hand, a critical reading model may serve as a useful point of reference to guide reviewers to employ more scrutiny on the parts of the paper appropriate for this specific paper type, which ultimately may improve reviewing quality. We assess that the opportunities provided by the introduced review assistance tasks outweigh the potential risks in general, yet highlight that a targeted study is necessary to substantiate this assessment.

References

2008–2023. Grobid. `https://github.com/kermitt2/grobid`.

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.

Lutz Bornmann. 2011. Scientific peer review. *Annual Review of Information Science and Technology*, 45(1):197–245.

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.

Liying Cheng, Lidong Bing, Qian Yu, Wei Lu, and Luo Si. 2020. APE: Argument pair extraction from peer review and rebuttal via multi-task learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7000–7011, Online. Association for Computational Linguistics.

Alexandra Chronopoulou, Matthew Peters, and Jesse Dodge. 2022. Efficient hierarchical domain adaptation for pretrained language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1336–1351, Seattle, United States. Association for Computational Linguistics.

Nils Dycke, Ilia Kuznetsov, and Iryna Gurevych. 2022. Yes-yes-yes: Proactive data collection for ACL rolling review and beyond. In *Findings of the Association for Computational Linguistics: Empirical Methods in Natural Language Processing 2022*, pages 300–318, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nils Dycke, Edwin Simpson, Ilia Kuznetsov, and Iryna Gurevych. 2021. Assisting decision making in scholarly peer review: A preference learning perspective. *arXiv:2109.01190*.

Raymond Fok, Hita Kambhamettu, Luca Soldaini, Jonathan Bragg, Kyle Lo, Marti Hearst, Andrew Head, and Daniel S Weld. 2023. Scim: Intelligent skimming support for scientific papers. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23, page 476–490, New York, NY, USA. Association for Computing Machinery.

Tirthankar Ghosal, Sandeep Kumar, Prabhat Kumar Bharti, and Asif Ekbal. 2022a. Peer review analyze: A novel benchmark resource for computational analysis of peer reviews. *PLOS ONE*, 17(1):1–29.

Tirthankar Ghosal, Kamal Kaushik Varanasi, and Valia Kordoni. 2022b. HedgePeer: A dataset for uncertainty detection in peer reviews. In *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries*, JCDL '22, pages 1–5, New York, NY, USA. Association for Computing Machinery.

Tirthankar Ghosal, Rajeev Verma, Asif Ekbal, and Pushpak Bhattacharyya. 2019. DeepSentiPeer: Harnessing Sentiment in Review Texts to Recommend Peer Review Decisions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1120–1130, Florence, Italy. Association for Computational Linguistics.

Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A. Smith, and Luke Zettlemoyer. 2022. DEMix layers: Disentangling domains for modular language modeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5557–5576, Seattle, United States. Association for Computational Linguistics.

Xinyu Hua, Mitko Nikolov, Nikhil Badugu, and Lu Wang. 2019. Argument mining for understanding peer reviews. 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 2131–2137, Minneapolis, Minnesota. Association for Computational Linguistics.

Rob Johnson, Anthony Watkinson, and Michael Mabe. 2018. *The STM Report: An overview of scientific and scholarly publishing*. International Association of Scientific, Technical and Medical Publishers, The Hague, Netherlands.

Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. A dataset of peer reviews (PeerRead): Collection, insights and NLP applications. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1647–1661, New Orleans, Louisiana. Association for Computational Linguistics.

Neha Kennard, Tim O'Gorman, Rajarshi Das, Akshay Sharma, Chhandak Bagchi, Matthew Clinton, Pranay Kumar Yelugam, Hamed Zamani, and Andrew McCallum. 2022. DISAPERE: A dataset for discourse structure in peer review discussions. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1234–1249, Seattle, United States. Association for Computational Linguistics.

Ilia Kuznetsov, Jan Buchmann, Max Eichler, and Iryna Gurevych. 2022. Revise and Resubmit: An Intertextual Model of Text-based Collaboration in Peer Review. *Computational Linguistics*, 48(4):949–986.

Carole J. Lee, Cassidy R. Sugimoto, Guo Zhang, and Blaise Cronin. 2013. Bias in peer review. *Journal of the American Society for Information Science and Technology*, 64(1):2–17.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning BERT: misconceptions, explanations, and strong baselines. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Barbara Plank. 2016. What to do about non-standard (or non-canonical) language in NLP. *arXiv:1608.07836*.

Simon Price and Peter A Flach. 2017. Computational support for academic peer review: A perspective from artificial intelligence. *Communications of the Association for Computing Machinery (ACM)*, 60(3):70–79.

Chenglei Qin and Chengzhi Zhang. 2022. Which structure of academic articles do referees pay more attention to?: perspective of peer review and full-text of academic articles. *CoRR*, abs/2209.01841.

Anna Rogers and Isabelle Augenstein. 2020. What can we do to improve peer review in NLP? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1256–1262, Online. ACL.

Nihar B Shah. 2019. Principled Methods to Improve Peer Review. Available online at `http://www.cs.cmu.edu/~nihars/publications/survey_peerreview_niharshah.pdf` (accessed on 22.01.2022).

Chenhui Shen, Liying Cheng, Ran Zhou, Lidong Bing, Yang You, and Luo Si. 2022. MReD: A meta-review dataset for structure-controllable text generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2521–2535, Dublin, Ireland. Association for Computational Linguistics.

Lukas Stappen, Georgios Rizos, Madina Hasan, Thomas Hain, and Björn W. Schuller. 2020. Uncertainty-aware machine support for paper reviewing on the interspeech 2019 submission corpus. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 1808–1812. ISCA.

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 Association for Computing Machinery (ACM) on Human-Computer Interaction*, 5:1–17.

Andrew Tomkins, Min Zhang, and William D. Heavlin. 2017. Reviewer bias in single- versus double-blind peer review. *Proceedings of the National Academy of Sciences*, 114(48):12708–12713.

Richard Walker and Pascal Rocha da Silva. 2015. Emerging trends in peer review—a survey. *Frontiers in Neuroscience*, 9.

Jingyan Wang and Nihar B. Shah. 2019. Your 2 is my 1, your 3 is my 9: Handling arbitrary miscalibrations in ratings. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*, pages 864–872. International Foundation for Autonomous Agents and Multiagent Systems.

Weizhe Yuan, Pengfei Liu, and Graham Neubig. 2022. Can we automate scientific reviewing? *Journal of Artificial Intelligence Research*, 75:171–212.

Rongsheng Zhang, Yinhe Zheng, Xiaoxi Mao, and Minlie Huang. 2021. Unsupervised domain adaptation with adapter. *arXiv:2111.00667*.

A Corpus Creation Details

A.1 Overview

Revision Matching Our data model assumes that each paper is associated with at least one revision – yet some of the existing peer review datasets only provide the submitted drafts. To remedy this, we augment ACL-17, CONLL-16 and COLING-20 with camera-ready versions by matching the accepted paper draft titles and abstracts against the ACL anthology[14].

For each of the papers in the mentioned datasets, we extracted the title and abstract either from the provided meta-data or the PDF. We then considered all papers of the respective conference in the ACL anthology and retrieved the top five entries according to the sentence BLEU[15] of the title and abstract with the paper at hand. Exact matches were included without manual verification, for the others the authors checked if the papers plausibly align. When in doubt, we opted to not include the matched camera-ready version.

Paper Parsing For all datasets except F1000-22, the raw paper inputs are PDFs, which are processed using GROBID (GRO, 2008–2023) and translated into the ITG data model (Kuznetsov et al., 2022) that captures the structural information (i.e. sections, subsections, etc.), linking information (i.e. citations and references) and layout information (i.e. lines and pages) of a document. For the datasets with line numbers in paper drafts (ACL-17, CONLL-16, ARR-22), we first remove line numbers for parsing and then match the line and page information heuristically to the processed papers. Although manual spot checks suggest a high quality of GROBID and ITG parsing, errors are inevitable and we provide papers both in raw and parsed form. To process F1000 Research XML papers, we extend and modify the existing parser provided in the ITG library.

Review Parsing The reviews are converted into a standardized format that supports structured and unstructured reviews, with and without scores. This means that for ARR-22, COLING-20, ACL-17, and CONLL-16 we include the rich set of review scores, and for ARR-22 the diverse review sections (strengths, weaknesses, etc.) remain intact.

[14] `https://aclanthology.org/`

[15] Using NLTK 3.7, see `https://www.nltk.org/`

Metadata and Versions NLPEER includes all available revisions of a paper. For F1000-22, each paper may have multiple revisions. Other datasets include at least the paper draft and if the paper was eventually published a camera-ready version of the paper. We associate rich metadata with each paper version, including the extracted title, abstract, authors (for accepted papers), and, if available, information on the paper type.

Personal data check To ensure that the published reviews pose no risks related to personal information, each peer review text in NLPEER – including the data adopted from prior work – was additionally validated by at least one NLP expert from our group. The initial analysis has identified 25 potentially problematic cases, of which 10 were deemed relevant upon a second check. Most of these cases contained either anonymous OpenReview.net identifiers within the review texts or included notes to the area chairs within the review main body. Despite low risk of cross-linking this information, we opted to discard these potentially privacy relevant sentences from the respective reviews.

A.2 COLING-2020 Collection

The data for the COLING-20 dataset was collected during the 28th International Conference on Computational Linguistics. Independent from the actual reviewing process, we asked authors and reviewers to donate their anonymized drafts and reviews, respectively.

Workflow After reviewing was completed, we reached out to authors and reviewers, asking them to consent to the research use of their data and to grant a CC0 license to their texts, with the the silence period of two years after COLING-20 acceptance decisions (October 2020). Reviewers and authors were informed about the risks of author profiling based on their provided textual artifacts. In total roughly 1500 anonymous reviews and 150 drafts were donated in this way. To adhere to the principles proposed by Dycke et al. (2022), we only include reviews and paper drafts for which both authors and reviewers agreed to donation; hereby avoiding any possibility of leaking confidential research ideas from the papers.

A.3 ARR-2022 Collection

ACL Rolling Review (ARR) is the unified and continuous reviewing system of the Association for Computational Linguistics (ACL). Papers are submitted in regular intervals, the *cycles*, receive reviews and a meta-review. In case of a positive meta-review, the paper becomes eligible for submission to any of the ACL conferences including the Annual meeting of the ACL and Empirical Methods for Natural Language Processing, where program chairs make the final acceptance decision. Otherwise, the paper is revised, resubmitted and typically reviewed by the same set of reviewers in a later cycle.

Workflow We followed the workflow with the exact same license transfer agreements proposed by Dycke et al. (2022) to collect peer reviewing data of the cycles September 2021 trough January 2022 covering papers later accepted at the annual meeting of the ACL (ACL 2022) and the annual meeting of the north-american chapter of the ACL (NAACL 2022). Independent from the reviewing process, reviewers were presented the option to donate all peer reviews of each cycle in bulk anytime during the reviewing period. After acceptance decisions for the conferences were released, we reached out to the authors of accepted papers roughly one month before the actual conference took place. Authors and reviewers were informed about the risks of author profiling and review release. The collected dataset contains reviews for the final draft of a paper, but none of the previous revisions. Some reviews included in the dataset are therefore revisions of previous reviews or may contain references to those.

A.4 F1000-22 Creation

F1000Research is an open post-publication reviewing platform covering articles from various fields, from clinical medicine to scientific policy research to R package development, as well as different article types, including case studies, literature reviews, research articles and code documentation. Publications are published *prior to acceptance*, and then can be approved, rejected or approved-with-reservations by one several invited reviewers. Publications can have multiple versions; each version is accompanied by open peer reviewing reports, author responses and amendment notes. All data on F1000Research is provided under an open license (CC-BY) in an easy-to-process JATS XML format.

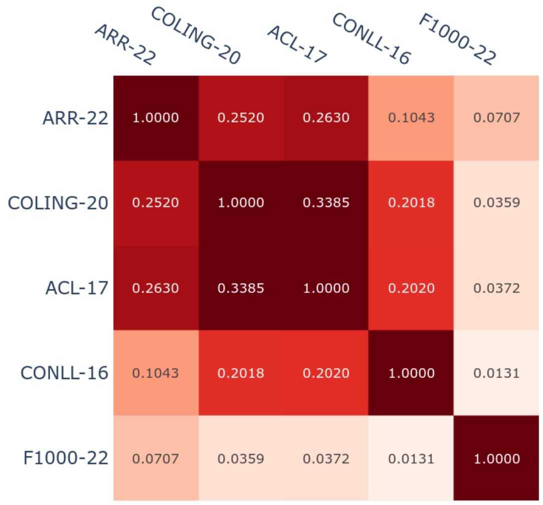

Figure 7: Vocabulary overlap on paper abstracts based on the Jaccard metric on lemmas of the datasets in NLPEER.

Workflow F1000 Research provides an official API for collecting peer reviews and articles[16]. We retrieved the index of articles and reviews in July 2022 and subsequently downloaded all articles with reviews for all versions and in XML, as well as PDF format. We discarded 34 articles with invalid file formatting and roughly 2000 articles that lacked reviews for the first version, as these indicate stale submissions. We extract meta-data, reviews and author responses from the article JATS XML files.

A.5 Extended Datasets Analysis

We complement the domain overlap analysis between reviews by the same analysis on paper abstracts. The vocabulary overlap in Figure 7 is generated under the same configuration (Jaccard metric on the top 10% of the lemmas), but computed on paper abstracts for all datasets. We see that the vocabulary overlap in abstracts is very similar to the overlap in review texts. However, the absolute similarity values are overall lower. This supports the observation that reviews have a wider cross-domain shared vocabulary, while papers apparently employ a more specialized register.

B Experiment Details

B.1 Fine-tuning Setup

The goal of our experiments is to identify the difficulty of each task and observe domain differences across datasets in NLPEER. We therefore fine-tune well-established large language models (LLMs) on each of the tasks; while the training objectives and approaches vary, we omit extensive fine-tuning of the LLMs and instead focus on a base set of hyper-parameters close to the recommended ones. During a pilot study we observed relatively unstable training, which rendered extensive random hyper-parameter search infeasible for the scope of this work.

	Learning Rate	Batch Size
RoBERTa	2.00×10^{-5}	16
BioBERT	1.00×10^{-5}	16
SciBERT	3.00×10^{-5}	32

Table 4: Hyper-paramters for fine-tuning the large language models consistent across tasks.

Fine-tuning Table 4 depicts the different fine-tuning parameters per LLM used for all experiments unless stated otherwise. We use the huggingface transformers implementation of RoBERTa[17] with roughly 125 million parameters, BioBERT[18] with around 110 million parameters, and SciBERT[19] with roughly 110 million parameters. We follow recommendations (Mosbach et al., 2021) for fine-tuning LLM on comparatively small datasets. For all of the models and datasets, we allow up to 20 epochs of training, employ a linear warm-up schedule (for 6% of the training steps) with non-bias weight decay of 0.1 and have 10 repeated measures on different random seeds to account for different random initializations of the task-specific classifier heads.

We implement the training and testing pipeline in pytorch lightning[20] using huggingface transformers[21]. For each run, we select the model with the best performance on the validation set at the end of each epoch. We implement an early stopping mechanism that stops fine-tuning if no improvement is observed after at most 8 epochs.

[16] https://f1000research.com/developers

[17] roberta-base, https://huggingface.co/roberta-base

[18] dmis-lab/biobert-v1.1, https://huggingface.co/dmis-lab/biobert-v1.1

[19] allenai/scibert_scivocab_uncased, https://huggingface.co/allenai/scibert_scivocab_uncased

[20] 1.7.7, https://www.pytorchlightning.ai/

[21] 4.22.2, https://huggingface.co/docs/transformers/index

Repeated Measures For each task and language model, we apply the fine-tuning procedure described above using the test and development sets. We repeat fine-tuning including model selection in total 10 times for each model and task. In each fine-tuning run we vary the random seed influencing the order of batches and randomly initialized weights of the model. We report the used random seeds within the code provided along the submission.

Stratified Splitting For the tasks review score prediction and guided skimming for peer review, we split the datasets with a special stratification criterion. For review score prediction, we require that the distribution of reviews per paper is similar across splits. For guided skimming, we make sure that the splits have a similar distribution of relevant paragraphs per paper. To achieve this, we employ sklearn's stratified, binary split function[22] while mapping the considered numerical stratification criterion to a discrete space by assigning the real numbers to buckets. To realize a split into three datasets, we realize repeated binary splits.

B.2 Pragmatic Labeling

For pragmatics labeling we map the semi-structured review form of ARR and the labels of the F1000RD dataset (Kuznetsov et al., 2022) to the same set of labels. The mapping of labels is summarized in Table 5. We highlight that, unlike the manually curated labels of F1000-RD, the sentences of ARR-22 are just extracted from the review forms which do not enforce full consistency with the respective section implying certain levels of noise in labels. For instance, some reviewers do mention strengths in the summary section of a review. Hence, our experiments are also targeted towards determining if this scalable approach to acquire a supervision signal is feasible; a further, detailed analysis of the quality of labels is a promising future direction of research.

B.3 Guided Skimming for Peer Review

We formulate guided skimming as a ranking task on the paragraphs of a paper considering their relevance to the writing of a peer review. To approach this task, we exploit explicit links from the review reports to the paper indicating that reviewers discuss a specific paragraph.

Explicit Link Detection Kuznetsov et al. (2022) propose a regular-expression-based algorithm to detect anchors (i.e. explicit mentions of structural elements in the paper) in review reports. The authors report an F1 score of 0.77 (with a precision of 0.81) for anchor identification and 0.64 (with a precision of 0.66) for their approach at matching explicit anchors to regions in the paper compared to human annotations. We conclude that the extraction of explicit links works reasonably well to be used as a proxy for reviewers' focus regions in the paper. Manual checks support this observation, in particular we see that explicit links seem to be identified with high precision, but comparatively low recall. Consequently, the resulting labels do contain certain levels of noise and reflect only a subset of the regions of the paper that are actually discussed in the reviews. We also highlight that implicit links in reviews, i.e. discussions of paper aspects that do not use explicit identifiers for paper regions, are not considered, as they are not readily available at scale.

Table 6 shows the regular expressions used for the extraction of explicit anchors in reviews. We extend the existing set of rules by line numbers and formulas as supported in ARR-22 and ACL-17. For matching these, we rely on the layout information extracted heuristically from the PDFs allowing a reliable mapping of paragraphs to line ranges. We aggregate the explicit links of all reviews for a paper to derive the set of focused paragraphs. We omit the frequency of links to paragraphs and consider only binary labels (linked or not-linked), to simplify the task and avoid making additional assumptions. While we in principle allow for any kind of explicit link (to paragraphs, sections, etc.) during extraction, we only include those that can be mapped to one specific paragraph leaving only quotes and line references in practice.

Relevant Paragraph Distribution within Papers In total the papers of ARR-22 have 12636 paragraphs of which 1100 (roughly 9%) are linked. In the ACL-17 dataset 600 of 4642 paragraphs (13%) are referenced by reviewers explicitly. We investigate the types of explicit links in each dataset. For ARR-22 64% of the relevant paragraphs are extracted from line-mentions in the reviews, while for ACL-17 these amount to 57%. Figures 8 and 9 show the histograms of text lengths for the relevant (i.e. linked) and not relevant (i.e. not linked) paragraphs of the papers. Overall, the length of relevant

[22] `train_test_split` of sklearn 0.0 https://scikit-learn.org

Label	ARR Review Section	ARR-22	F10000-RD	F1000-22
Neutral	Summary	3466	Recap, Other, Structure	1932
Strength	Strengths	2565	Strength	415
Weakness	Weaknesses	4458	Weaknesses	811
Request	Questions and Requests	4257	Todo	1514
		14746		**4672**

Table 5: Pragmatic label mapping unifying F1000-RD and the ARR review form sections. The column ARR-22 and F1000-22 indicate the number of sentences with the given label within the respective datasets. The bottom row shows the total amount of sentences within each of the datasets.

paragraphs tends to be higher than that of short paragraphs. We suspect that this phenomenon is encouraged by two interacting factors: first, paper parses are not perfect especially for splitting PDFs into structural elements leading to differently sized paragraphs in addition to naturally occurring size variations. Second, in combination with the first factor, longer paragraphs are simply more likely to be linked, because they have more content that can be discussed. This encourages a further line of work based on length-normalized paragraphs; we also investigate paragraph length as a spurious feature for the skimming task in the following.

Further on, we investigate the position of relevant paragraphs within the document hierarchy (sections, subsections, etc.) of the papers. For ARR-22 around half of the paragraphs originate from the level of sections rather than sub-sections, while for ACL-17 roughly two thirds lie on section level. This suggests no substantial bias towards a level in the document hierarchy.

Finally, we analyse the distribution of linked paragraphs within a set of canonical sections typical for NLP papers, including e.g. introduction, method, and results. Figure 10 and 11 show the frequency of links to each of these canonical section types. We map non-canonical section titles to the "other" category. In both datasets the number of explicit links pointing to the introduction is the highest when ignoring the section type "other". For ARR-22 this phenomenon is less pronounced, while the results section is slightly more often referenced than in ACL-17. Overall, there appear to exist natural areas of focus for reviewers as approximated by explicit links, but paragraphs of many different section types are in fact covered in both datasets. As our baseline models don't take structural information into account the observed light skews towards certain sections is unlikely to be the primary explanation for the non-trivial performance of the LLM reported in Section 5.4.

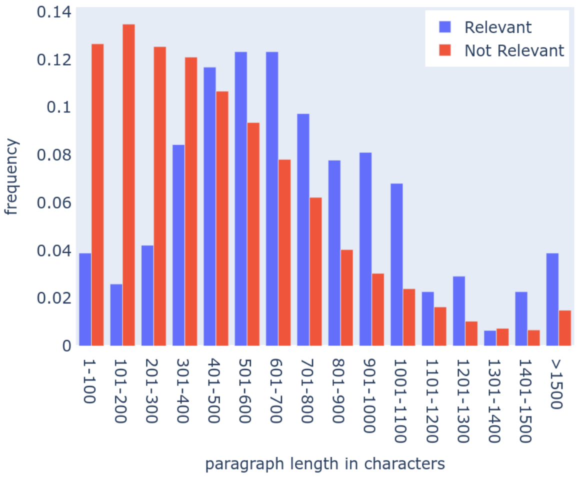

Figure 8: Distribution of paragraph lengths for ACL-17, where "relevant" marks paragraphs that were in fact referenced by reviewers.

Type	Rule	Example
fig-ix	[Ff]igure | [Ff]ig(\.?)	*Figure 3 shows...*
table-ix	[Tt]able (\.?)	*I don’t understand [...] mentioned in table 1.*
sec-ix	[Ss]ection | [Ss]ec (\.?)	*There is a typo in the title of sec 3.*
sec-name	(?P<ix>[A-Z][A-z]+) | [“"’](?P<ix>[A-Z][A-z]+)[”"’] [Ss]ection | (?P<ix>\b[Tt]itle\b)	*In the "Methods" section [...].*
quote	*omitted due to excessive length – including various forms of quotation marks*	*The authors claim "...".*
ref-ix	[Rr]ef(|erence)? ?(?P<ix>\d+)	*ref 23 is mal-formatted.*
page	(\bp\.?|page)	*On page 1 [...].*
paragraph	\b(pp\.?|paragraph|para\.?) ?(?P<ix>\d+)	*paragraph 3 introduces [...].*
line	\b(line|l\.?) ?(?P<ix>\d+) | (?P<ix>first) | (?P<ix>second) | (?P<ix>third) | [IiOo]n lines? (?P<ix>\d+) | [IiOo]n l\.? (?P<ix>\d+) | [IiOo]n ll\.? (?P<ix>\d+) | (?P<ix>)(st|nd|rd)	*On lines 23-28 [...]*
page	\b(?P<ix>\d)(st|nd|rd) | (?P<ix>first) | (?P<ix>second) | (?P<ix>third)	*On page 7 [...].*
formula	\b[Ee]q(uations?|?) ?(?P<ix>\d+)	*There is an error in equation 3*

Table 6: Selection of the regular expressions used for identifying explicit anchors in review reports. The horizontal line separates the rules proposed by Kuznetsov et al. (2022) from the extended rules introduced in this work. The full set of rules is provided in the code associated with this work. For each type of explicit link, we provide an example.

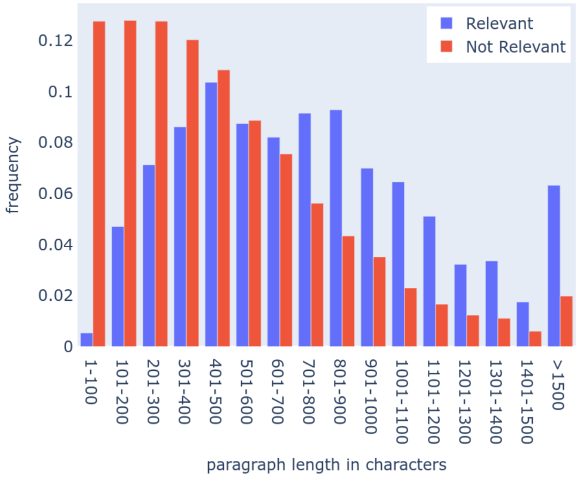

Figure 9: Distribution of paragraph lengths for ARR-22, where "relevant" marks paragraphs that were in fact referenced by reviewers.

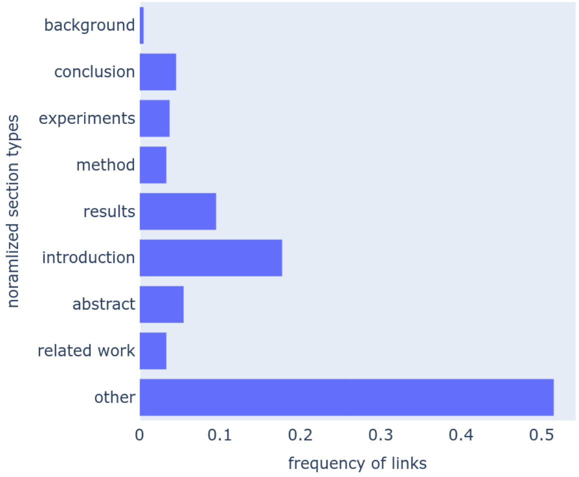

Figure 10: Distribution of links by normalized section type for ARR-22. Sections with non-standard names are mapped to the other category.

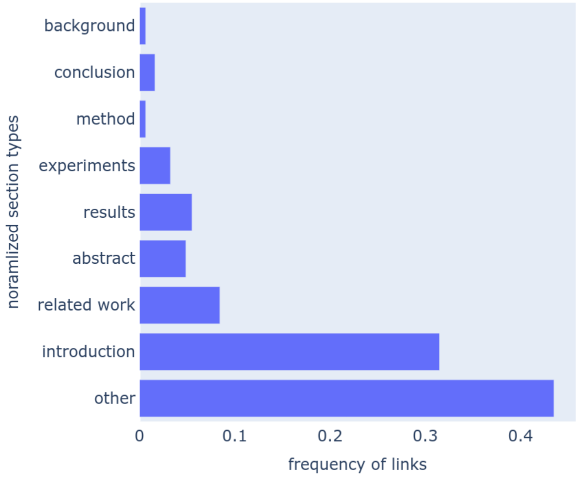

Figure 11: Distribution of links by normalized section type for ACL-17. Sections with non-standard names are mapped to the other category.

C Detailed Results

C.1 Review Score Prediction

We report the complete and more detailed results of the review score prediction models including all tested large language models (RoBERTa, BioBERT, SciBERT) and the mean score baseline in Table 7.

Additional Metrics All metrics are computed on the actual review score scale; hence model outputs are mapped back from the normalized scale. In addition to the mean root squared error (MRSE) and the F1-macro on discretized output predictions, we report the R^2 metric to measure the degree of fit of the regression models, as well as an additional score diversity criterion. For this criterion, we compute the overall score distribution for a model across all samples of the test set and compare it with the true, human score distribution (using KL-divergence) to measure whether the diversity of model scores aligns with human reviewers. A model that learns reviewers' rating behavior well, should also predict scores from a similar range of scores as humans do. Models over-focusing or ignoring single scores would have high scores; hence lower is better in this case.

KL-Divergence Interpretation We see that the mean overall score consistently shows the highest scores meaning the lowest similarity to the human score distribution. While the best model according to MRSE tends to have one of the lowest scores, this is not consistently true (e.g. RoBERTa on F1000-22). For COLING-20 and CONLL-16 the models perform on-par with mean baseline suggesting that they converged to predicting scores very close to the mean. Overall, even the best models still show notably different rating behavior from humans across all samples and datasets.

C.2 Pragmatic Labeling

We extend the results presented in the main body of the paper by a concise error analysis for the within and out-of dataset experiments. In the following report the confusion matrices for the best performing LLM (RoBERTa) of the model that achieved the reported median performance.

Within-dataset Errors Table 8 shows the confusion matrix for RoBERTa on the F1000-RD dataset. We see the highest model confusion for the neutral class; strengths, weaknesses and requests are all most commonly confused with a neutral statement

	MRSE ↓	R^2 ↑	**F1-macro** ↑	**Score KL-Divergence** ↓
ARR-22				
RoBERTa	0.37 ± 0.02	0.16 ± 0.05	0.46 ± 0.04	0.28 ± 0.11
BioBERT	0.45 ± 0.03	−0.03 ± 0.07	0.29 ± 0.06	0.71 ± 0.33
SciBERT	0.38 ± 0.03	0.13 ± 0.08	0.40 ± 0.09	0.45 ± −1.00
mean score	0.45	−0.02	0.24	2.21
COLING-20				
RoBERTa	0.45 ± 0.08	−0.33 ± 0.24	0.15 ± 0.01	1.74 ± 0.07
BioBERT	0.44 ± 0.09	−0.31 ± 0.27	0.15 ± 0.07	1.74 ± 0.17
SciBERT	0.47 ± 0.06	−0.38 ± 0.16	0.15 ± 0.01	1.74 ± 0.09
mean score	0.22	−0.06	0.23	1.74
ACL-17				
RoBERTa	0.79 ± 0.04	−0.02 ± 0.05	0.06 ± 0.01	2.38 ± 0.07
BioBERT	0.76 ± 0.06	0.02 ± 0.07	0.11 ± 0.04	1.76 ± 0.34
SciBERT	0.78 ± 0.07	−0.00 ± 0.09	0.08 ± 0.04	2.31 ± 0.42
mean score	0.78	0	0.06	2.38
CONLL-16				
RoBERTa	0.77 ± 0.04	0.01 ± 0.05	0.08 ± 0.00	2.81 ± 0.00
BioBERT	0.87 ± 0.04	−0.12 ± 0.05	0.08 ± 0.01	2.81 ± 0.41
SciBERT	0.89 ± 0.29	−0.15 ± 0.38	0.08 ± 0.12	2.81 ± −1.00
mean score	0.78	0	0.08	2.81
F1000-22				
RoBERTa	0.20 ± 0.01	0.47 ± 0.03	0.38 ± 0.09	0.75 ± 0.29
BioBERT	0.21 ± 0.01	0.44 ± 0.02	0.41 ± 0.04	0.60 ± 0.10
SciBERT	0.21 ± 0.01	0.45 ± 0.02	0.40 ± 0.06	0.65 ± 0.18
mean score	0.41	0	0.19	1.28

Table 7: MRSE is measured on non-normalized scores (i.e. absolute values cannot be compared across datasets); here, lower is better. F1-macro computed on discretized scores, where higher is better. KL-Divergence is computed relative to the human score distribution across all samples of each dataset; hence, a lower value indicates a more similar distribution compared to humans. We report the median score over ten runs and provide the standard deviation.

and vice-versa. We highlight that in our experiments the class neutral subsumes the more fine-grained neutral labels of F1000-RD like `summary` or `structure`, which might be one factor contributing to harded delineation.

In Table 9 we report the confusion matrix for RoBERTa on the ARR-22 dataset. Generally, we seem a similar pattern as for F1000-RD: the neutral class is most often confused with the others. As the review forms are not strictly enforced, it is likely that strengths and weaknesses are already reported in the summary section correlating to the `neutral` label encouraging this confusion. Likewise several neutral, factual sentences exist in the strength and weaknesses sections. This shows one limitation of using structure review forms as a proxy for review sentence pragmatic. Interestingly, we see that requests and weaknesses are very commonly confused. While the noisy supervision labels might again be a contributing factor, this aligns with the reported highest human disagreement for these two classes by Kuznetsov et al. (2022).

Out-of-dataset Errors We inspect the errors of the best performing models trained on ARR-22 and tested on F1000-RD, and vice versa. Table 10 reports the confusion matrix for the first case. We see that unlike the in-domain trained model on F1000-RD weaknesses are frequently confused with requests and vice versa. This confirms that the request and weaknesses sections in the ARR review forms lead to the most ambiguity in the supervision labels as already hypothesized in the previous

	strength	weakness	request	neutral
strength	63	1	0	7
weakness	3	144	5	14
request	4	6	270	6
neutral	11	31	23	346

Table 8: Confusion matrix of RoBERTa on F1000-RD. The entry in row i and column j denotes that the label j was predicted by the model for a sample of true class i.

	strength	weakness	request	neutral
strength	356	30	13	103
weakness	29	579	207	59
request	30	254	565	22
neutral	49	60	11	581

Table 9: Confusion matrix of RoBERTa on ARR-22. The entry in row i and column j denotes that the label j was predicted by the model for a sample of true class i.

paragraph. Similar to the in-domain trained model the neutral class is the hardest to predict for the model. Overall, the transfer performance lies in a promising range that suggests more efforts on few-shot and cross-domain transfer of models are good future directions of research.

The transfer of a model trained on F1000-RD to ARR-22 aligns with previous observations for the within evaluation on ARR-22: most confusion is observed for the neutral class. Additionally, the model predicts the class `request` very frequently for sentences belonging to `weaknesses`. This supports the hypothesis that many sentences in the weakness section of ARR are actually requests, as the model trained on F1000-RD is based off of human gold annotations.

C.3 Guided Skimming for Peer Review

In addition to the at-k-measures and aggregate metrics of the skimming performance, as reported in the main body of this work, we provide the full results of all models and additional metrics in this section.

Paragraph Length Baseline As reported in B.3, the length distribution of linked and non-linked paragraphs might be a useful spurious feature for ranking the paragraphs by relevance to the guided skimming process. While the truncation of model inputs of the used large language models to 512 tokens makes is unlikely that the models are at risk of exploiting sequence length to achieve non-trivial performance, future approaches using, for instance, the full paper text or full paragraphs might do so. Hence, in the following we also report the performance of the baseline that ranks paragraphs by their number of characters.

Ranking Measures Table 12 reports the performance of the LLMs and the baselines on ACL-17. We consider the mean reciprocal rank (MRR) and the area under the receiver operating characteristic curve (AUROC) as overall measures of the ranking quality in addition to the at-k-measures. For ACL-17, SciBERT shows the best performance according to MRR, AUROC and Precision@3. While the model performs substantially above the random baseline, the margin towards the length baseline is small, especially for the AUROC. Overall, the ranking produced by SciBERT leads to a higher precision in the top ranks, but seemingly performs on-par with the length baseline for the overall ranking performance. This shows the difficulty of the task at hand, but at the same time suggests that there is a useful training signal in the paragraph texts beyond their mere length.

The results on ARR-22 as shown in 13 are very similar. Although all models perform above random, the paragraph length baseline is hard to beat. Here, we can observe some improvements in terms of recall and precision in the top ranks, but the overall ranking performance lies below the length baseline.

Conclusion The length of the paragraphs seems a relevant feature that might be exploited as a spu-

	strength	weakness	request	neutral
strength	54	7	2	8
weakness	4	106	55	1
request	2	75	209	0
neutral	33	171	127	80

Table 10: Confusion matrix of RoBERTa trained on ARR-22 and transferred to F1000-RD. The entry in row i and column j denotes that the label j was predicted by the model for a sample of true class i.

	strength	weakness	request	neutral
strength	299	9	19	175
weakness	38	295	290	251
request	30	160	451	230
neutral	65	32	26	578

Table 11: Confusion matrix of RoBERTa trained on F1000-RD and transferred to ARR-22. The entry in row i and column j denotes that the label j was predicted by the model for a sample of true class i.

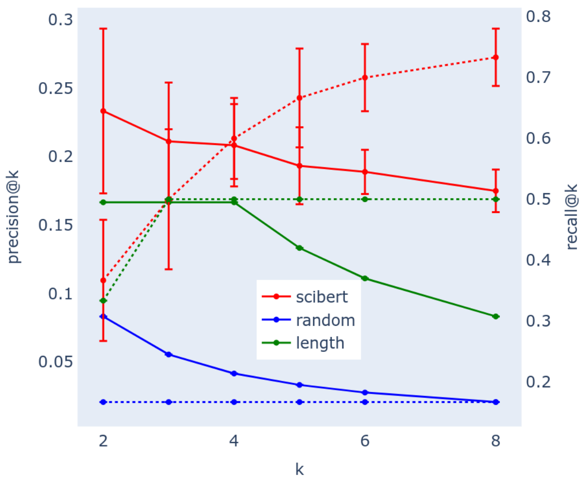

Figure 12: Precision (left) and recall (right, dotted lines) at different ranks k for SciBERT, random and length baseline on ACL-17.

rious decision criterion by the models. However, especially in the top ranks for ACL-17 (see 12) the LLM seem to pick up information beyond mere paragraph length. We suspect that structural and contextual information would be beneficial for this hard task that would increase the margin towards the paragraph length baseline. Additionally, more elaborate training regimes considering list-wise losses a interesting future directions. To eliminate the risk of learning paragraph length as a spurious pattern the normalization of paragraph lengths by different segmentation techniques is promising.

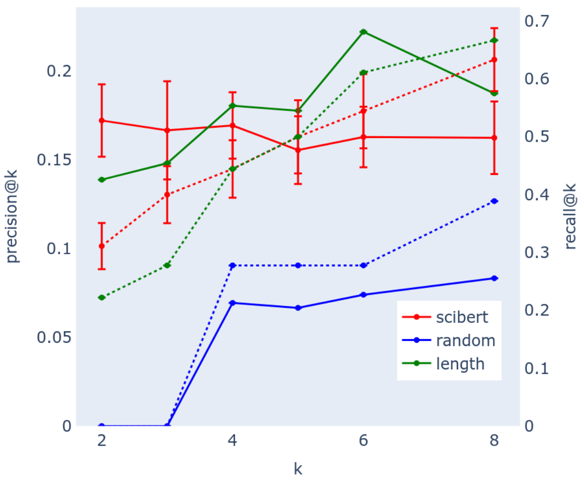

Figure 13: Precision (left) and recall (right, dotted lines) at different ranks k for SciBERT, random and length baseline on ARR-22.

	MRR ↑	**AUROC ↑**	**Precision @ 3 ↑**	**Recall @ 3 ↑**
RoBERTa	0.46 ± 0.05	0.73 ± 0.06	0.22 ± 0.03	0.53 ± 0.05
BioBERT	0.40 ± 0.11	0.76 ± 0.06	0.19 ± 0.05	0.47 ± 0.08
SciBERT	0.43 ± 0.06	0.77 ± 0.04	0.21 ± 0.04	0.50 ± 0.12
Length Baseline	0.34 ± 0.00	0.76 ± 0.00	0.17 ± 0.00	0.50 ± 0.00
Random Baseline	0.14 ± 0.05	0.47 ± 0.06	0.08 ± 0.03	0.20 ± 0.08

Table 12: Performance on ACL-17. Mean Reciprocal Rank (MRR), area under the receiver operating characteristic curve (AUROC) and precision and recall at $k = 3$ reported for the LLM, length baseline and random baseline. For all metrics, higher is better.

	MRR ↑	**AUROC ↑**	**Precision @ 3 ↑**	**Recall @ 3 ↑**
RoBERTa	0.30 ± 0.04	0.68 ± 0.03	0.15 ± 0.03	0.33 ± 0.05
BioBERT	0.35 ± 0.02	0.69 ± 0.03	0.17 ± 0.03	0.38 ± 0.04
SciBERT	0.35 ± 0.03	0.70 ± 0.02	0.17 ± 0.03	0.40 ± 0.05
Length Baseline	0.31 ± 0.00	0.75 ± 0.00	0.15 ± 0.00	0.28 ± 0.00
Random Baseline	0.18 ± 0.04	0.49 ± 0.02	0.07 ± 0.02	0.20 ± 0.05

Table 13: Performance on ARR-22. Mean Reciprocal Rank (MRR), area under the receiver operating characteristic curve (AUROC) and precision and recall at $k = 3$ reported for the LLM, length baseline and random baseline. For all metrics, higher is better.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, in the Limitations section.

☑ A2. Did you discuss any potential risks of your work?
Yes, in the Conclusions and Intended Use, as well as the Applications and Limitations sections.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☑ A4. Have you used AI writing assistants when working on this paper?
Grammar correction and spell checking

B ☑ Did you use or create scientific artifacts?

3

☑ B1. Did you cite the creators of artifacts you used?
Yes. Moreover, we attribute the individual content whenever possible or requested by the text authors.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
3

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
6

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
3.3

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Yes, to the extent possible, given the anonymity of parts of the data. Section 3.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Yes, Table 1 gives an overview over all data and Appendix B details the different experimental data configurations

C ☑ Did you run computational experiments?

5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Experiments Section and Appendix B

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Experiments Section and Appendix B.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix B

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.