The Diminishing Returns of Masked Language Models to Science

Zhi Hong[*], **Aswathy Ajith**[*], **J. Gregory Pauloski**[*], **Eamon Duede**[†], **Kyle Chard**[*‡], **Ian Foster**[*‡]

[*]Department of Computer Science, University of Chicago, Chicago, IL 60637, USA
[†]Department of Philosophy and Committee on Conceptual and Historical Studies of Science,
University of Chicago, Chicago, IL 60637, USA
[‡]Data Science and Learning Division, Argonne National Laboratory, Lemont, IL 60615, USA

Abstract

Transformer-based masked language models such as BERT, trained on general corpora, have shown impressive performance on downstream tasks. It has also been demonstrated that the downstream task performance of such models can be improved by pretraining larger models for longer on more data. In this work, we empirically evaluate the extent to which these results extend to tasks in science. We use 14 domain-specific transformer-based models (including SCHOLARBERT, a new 770M-parameter science-focused masked language model pretrained on up to 225B tokens) to evaluate the impact of training data, model size, pretraining and finetuning time on 12 downstream scientific tasks. Interestingly, we find that increasing model size, training data, or compute time does not always lead to significant improvements (i.e., $> 1\%$ F1), if any, in scientific information extraction tasks. We offer possible explanations for this surprising result.

1 Introduction

Massive growth in the number of scientific publications places considerable cognitive burden on researchers (Teplitskiy et al., 2022). Language models can potentially alleviate this burden by automating the scientific knowledge extraction process. BERT (Devlin et al., 2019) was pretrained on a general corpus (BooksCorpus and Wikipedia) which differs from scientific literature in terms of the context, terminology, and writing style (Ahmad, 2012). Other masked language models have since been pretrained on domain-specific scientific corpora (Huang and Cole, 2022; Gu et al., 2021; Gururangan et al., 2020; Beltagy et al., 2019) with the goal of improving downstream task performance. (We use the term *domain* to indicate a specific scientific discipline, such as biomedical science or computer science.) Other studies (Liu et al., 2019; Kaplan et al., 2020) explored the impact of varying model size, training corpus size, and compute time on downstream task performance. However, no previous work has investigated how these parameters affect science-focused models.

In this study, we train a set of scientific language models of different sizes, collectively called SCHOLARBERT, on a large, multidisciplinary scientific corpus of 225B tokens to understand the effects of model size, data size, and compute time (specifically, pretraining and finetuning epochs) on downstream task performance. We find that for information extraction tasks, an important application for scientific language models, the performance gains by training a larger model for longer with more data are not robust—they are highly task-dependent. We make the SCHOLARBERT models and a sample of the training corpus publicly available to encourage further studies.

2 Related Work

Prior research has explored the effects of varying model size, dataset size, and amount of compute on language model performance.

Kaplan et al. (2020) demonstrated that cross-entropy training loss scales as a power law with model size, dataset size, and compute time for unidirectional decoder-only architectures. Brown et al. (2020) showed that language model few-shot learning abilities can be improved by using larger models. However, both studies explored only the Generative Pretrained Transformer (GPT), an autoregressive generative model (Brown et al., 2020).

Comparing BERT-Base (110M parameters) and BERT-Large (340M parameters), Devlin et al. (2019) showed that masked language models can also benefit from more parameters. Likewise, Liu et al. (2019) demonstrate how BERT models can benefit from training for longer periods, with bigger batches, and with more data.

Models such as BERT and RoBERTa were pretrained on general corpora. To boost performance on scientific downstream tasks, SciBERT (Belt-

Findings of the Association for Computational Linguistics: ACL 2023, pages 1270–1283
July 9-14, 2023 ©2023 Association for Computational Linguistics

agy et al., 2019), PubMedBERT (Gu et al., 2021), BioBERT (Lee et al., 2020), and MatBERT (Trewartha et al., 2022) were trained on domain-specific text with the goal of enhancing performance on tasks requiring domain knowledge. Yet there is no work on how that task performance varies with pretraining parameters.

3 Data and Methodology

We outline the pretraining dataset, related models to which we compare performance, and the architecture and pretraining process used for creating the SCHOLARBERT models.

3.1 The Public Resource Dataset

We pretrain the SCHOLARBERT models on a dataset provided by Public.Resource.Org, Inc. ("Public Resource"), a nonprofit organization based in California. This dataset was constructed from a corpus of 85M journal article PDF files, from which the Grobid tool, version 0.5.5, was used to extract text (GROBID). Not all extractions were successful, because of corrupted or badly encoded PDF files. We work here with text from ∼75M articles in this dataset, categorized as 45.3% biomedicine, 23.1% technology, 20.0% physical sciences, 8.4% social sciences, and 3.1% arts & humanities. (A sample of the extracted texts and corresponding original PDFs is available in the Data attachment for review purposes.)

3.2 Models

We consider 14 BERT models: seven from existing literature (BERT-Base, BERT-Large, SciBERT, PubMedBERT, BioBERT v1.2, MatBERT, and BatteryBERT: Appendix A); and seven SCHOLARBERT variants pretrained on different subsets of the Public Resource dataset (and, in some cases, also the WikiBooks corpus). We distinguish these models along the four dimensions listed in Table 1: architecture, pretraining method, pretraining corpus, and casing. SCHOLARBERT and SCHOLARBERT-XL, with 340M and 770M parameters, respectively, are the largest science-specific BERT models reported to date. Prior literature demonstrates the efficacy of pretraining BERT models on domain-specific corpora (Sun et al., 2019; Fabien et al., 2020). However, the ever-larger scientific literature makes pretraining domain-specific language models prohibitively expensive. A promising alternative is to create larger, multi-disciplinary BERT models, such as SCHOLARBERT, that harness the increased availability of diverse pretraining text; researchers can then adapt (i.e., finetune) these general-purpose science models to meet their specific needs.

3.3 SCHOLARBERT Pretraining

We randomly sample 1%, 10%, and 100% of the Public Resource dataset to create PRD_1, PRD_10, and PRD_100. We pretrain SCHOLARBERT models on these PRD subsets by using the RoBERTa pretraining procedure, which has been shown to produce better downstream task performance in a variety of domains (Liu et al., 2019). See Appendix B.2 for details.

4 Experimental Results

We first perform sensitivity analysis across ScholarBERT pretraining dimensions to determine the trade-off between time spent in pretraining versus finetuning. We also compare the downstream task performance of SCHOLARBERT to that achieved with other BERT models. Details of each evaluation task are in Appendix C.

4.1 Sensitivity Analysis

We save checkpoints periodically while pretraining each SCHOLARBERT(-XL) model. In this analysis, we checkpoint at ∼0.9k, 5k, 10k, 23k, and 33k iterations based on the decrease of training loss between iterations. We observe that pretraining loss decreases rapidly until around 10 000 iterations, and that further training to convergence (roughly 33 000 iterations) yields only small decreases of training loss: see Figure 1 in Appendix.

To measure how downstream task performance is impacted by pretraining and finetuning time, we finetune each of the checkpointed models for 5 and 75 epochs. We observe that: (1) The undertrained 0.9k-iteration model sees the biggest boost in the F1 scores of downstream tasks (+8%) with more finetuning, but even with 75 epochs of finetuning the 0.9k-iteration models' average F1 score is still 19.9 percentage points less than that of the 33k-iteration model with 5 epochs of finetuning. (2) For subsequent checkpoints, the performance gains from more finetuning decreases as the number of pretraining iterations increases. The average downstream task performance of the 33k-iteration model is only 0.39 percentage points higher with 75 epochs of finetuning than with 5 epochs. There-

Model	Architecture	Pretraining Method	Casing	Pretraining Corpus	Domain	Tokens
BERT_Base	BERT-Base	BERT	Cased	Wiki + Books	Gen	3.3B
SciBERT	BERT-Base	BERT	Cased	SemSchol	Bio, CS	3.1B
PubMedBERT	BERT-Base	BERT	Uncased	PubMed_A + PMC	Bio	16.8B
BioBERT_1.2	BERT-Base	BERT	Cased	PubMed_B + Wiki + Books	Bio, Gen	7.8B
MatBERT	BERT-Base	BERT	Cased	MatSci	Mat	8.8B
BatteryBERT	BERT-Base	BERT	Cased	Battery	Mat	5.2B
BERT_Large	BERT-Large	BERT	Cased	Wiki + Books	Gen	3.3B
ScholarBERT_1	BERT-Large	RoBERTa-like	Cased	PRD_1	Sci	2.2B
ScholarBERT_10	BERT-Large	RoBERTa-like	Cased	PRD_10	Sci	22B
ScholarBERT_100	BERT-Large	RoBERTa-like	Cased	PRD_100	Sci	221B
ScholarBERT_10_WB	BERT-Large	RoBERTa-like	Cased	PRD_10 + Wiki + Books	Sci, Gen	25.3B
ScholarBERT_100_WB	BERT-Large	RoBERTa-like	Cased	PRD_100 + Wiki + Books	Sci, Gen	224.3B
ScholarBERT-XL_1	BERT-XL	RoBERTa-like	Cased	PRD_1	Sci	2.2B
ScholarBERT-XL_100	BERT-XL	RoBERTa-like	Cased	PRD_100	Sci	221B

Table 1: Characteristics of the 14 BERT models considered in this study. The BERT-Base and -Large architectures are described in (Devlin et al., 2019); the BERT-XL architecture has 36 layers, hidden size of 1280, and 20 heads. Details of the pretraining corpora are in Table 4 in the Appendix. The domains are Bio=biomedicine, CS=computer science, Gen=general, Mat=materials science and engineering, and Sci=broad scientific.

fore, in the remaining experiments, we use the ScholarBERT(-XL) model that was pretrained for 33k iterations and finetuned for 5 epochs.

4.2 Finetuning

We finetuned the ScholarBERT models and the state-of-the-art scientific models listed in Table 1 on NER, relation extraction, and sentence classification tasks. F1 scores for each model-task pair, averaged over five runs, are shown in Tables 2 and 3. For NER tasks, we use the CoNLL NER evaluation Perl script (Sang and De Meulder, 2003) to compute F1 scores for each test.

Tables 2 and 3 show the results, from which we observe: (1) With the same training data, a larger model does not always achieve significant performance improvements. BERT-Base achieved F1 scores within 1 percentage point of BERT-Large on 6/12 tasks; SB_1 achieved F1 scores within 1 percentage point of SB-XL_1 on 7/12 tasks; SB_100 achieved F1 scores within 1 percentage point of SB-XL_100 on 6/12 tasks. (2) With the same model size, a model pretrained on more data cannot guarantee significant performance improvements. SB_1 achieved F1 scores within 1 percentage point of SB_100 on 8/12 tasks; SB_10_WB achieved F1 scores within 1 percentage point of SB_100_WB on 7/12 tasks; SB-XL_1 achieved F1 scores within 1 percentage point of SB-XL_100 on 10/12 tasks. (3) Domain-specific pretraining cannot guarantee significant performance improvements. The Biomedical domain is the only domain where we see the on-domain model (i.e., pretrained for the associated domain; marked with underlines; in this case, PubMedBERT) consistently outperformed models pretrained on off-domain or more general corpora by more than 1 percentage point F1. The same cannot be said for CS, Materials, or Multi-Domain tasks.

4.3 Discussion

Here we offer possible explanations for the three observations above. (1) The nature of the task is more indicative of task performance than the size of the model. In particular, with the same training data, a larger model size impacts performance only for relation extraction tasks, which consistently saw F1 scores increase by more than 1 percentage point when going from smaller models to larger models (i.e., BERT-Base to BERT-Large, SB_1 to SB-XL_1, SB_100 to SB-XL_100). In contrast, the NER and sentence classification tasks did not see such consistent significant improvements. (2) Our biggest model, ScholarBERT-XL, is only twice as large as the original BERT-Large, but its pretraining corpus is 100X larger. The training loss of the ScholarBERT-XL_100 model dropped rapidly only in the first ∼10k iterations (Fig. 1 in Appendix), which covered the first 1/3 of the PRD corpus, thus it is possible that the PRD corpus can saturate even our biggest model. (Kaplan et al., 2020; Hoffmann et al., 2022). (3) Finetuning can compensate for missing domain-specific knowledge in pretraining data. While pretraining language models on a specific domain can help learn domain-specific concepts, finetuning can also fill holes in the pretraining corpora's domain knowledge, as long as the pretraining corpus incorporates

Domain	Biomedical				CS	Materials	Multi-Domain	Sociology	
Dataset	BC5CDR	JNLPBA	NCBI-Disease	ChemDNER	SciERC	MatSciNER	ScienceExam	Coleridge	Mean
BERT-Base	85.36	72.15	84.28	84.84	56.73	78.51	78.37	57.75	74.75
BERT-Large	86.86	72.80	84.91	85.83	59.20	82.16	82.32	57.46	76.44
SciBERT	88.43	73.24	86.95	85.76	**59.36**	82.64	78.83	54.07	76.16
PubMedBERT	**89.34**	**74.53**	**87.91**	**87.96**	59.03	82.63	69.73	57.71	76.11
BioBERT	88.01	73.09	87.84	85.53	58.24	81.76	78.60	57.04	76.26
MatBERT	86.44	72.56	84.94	86.09	58.52	**83.35**	80.01	56.91	76.10
BatteryBERT	87.42	72.78	87.04	86.49	59.00	82.94	78.14	**59.87**	**76.71**
SB_1	87.27	73.06	85.49	85.25	58.62	80.87	82.75	55.34	76.08
SB_10	87.69	73.03	85.65	85.80	58.39	80.61	**83.24**	53.41	75.98
SB_100	87.84	73.47	85.92	85.90	58.37	82.09	83.12	54.93	76.46
SB_10_WB	86.68	72.67	84.51	83.94	57.34	78.98	83.00	54.29	75.18
SB_100_WB	86.89	73.16	84.88	84.31	58.43	80.84	82.43	54.00	75.62
SB-XL_1	87.09	73.14	84.61	85.81	58.45	82.84	81.09	55.94	76.12
SB-XL_100	87.46	73.25	84.73	85.73	57.26	81.75	80.72	54.54	75.68

Table 2: NER F1 scores for each model. Models are finetuned five times for each dataset and the average result is presented. Underlined results represent the F1-scores of models trained on in-distribution data for the given task, and bolded results indicate the best performing model on that task. SB = SCHOLARBERT.

Domain	CS	Biomedical	Multi-Domain	Materials	
Dataset	SciERC	ChemProt	PaperField	Battery	Mean
BERT-Base	74.95	83.70	72.83	96.31	81.95
BERT-Large	80.14	88.06	73.12	**96.90**	84.56
SciBERT	79.26	89.80	73.19	96.38	84.66
PubMedBERT	77.45	**91.78**	**73.93**	96.58	**84.94**
BioBERT	80.12	89.27	73.07	96.06	84.63
MatBERT	79.85	88.15	71.50	96.33	83.96
BatteryBERT	78.14	88.33	73.28	96.06	83.95
SB_1	73.01	83.04	72.77	94.67	80.87
SB_10	75.95	82.92	72.94	92.83	81.16
SB_100	76.19	87.60	73.14	92.38	82.33
SB_10_WB	73.17	81.48	72.37	93.15	80.04
SB_100_WB	76.71	83.98	72.29	95.55	82.13
SB-XL_1	74.85	90.60	73.22	88.75	81.86
SB-XL_100	**80.99**	89.18	73.66	95.44	84.82

Table 3: F1 scores for each model on Relation Extraction (SciERC, ChemProt) and Sentence Classification (PaperField, Battery) tasks. Models are finetuned five times for each dataset and the average result is presented. Underlined results represent the F1-scores of models trained on in-distribution data for the given task, and bolded results indicate the best performing model on that task. SB = SCHOLARBERT.

the characteristics specific to the finetuning dataset.

5 Conclusions

We have reported experiments that compare and evaluate the impact of various parameters (model size, pretraining dataset size and breadth, and pretraining and finetuning lengths) on the performance of different language models pretrained on scientific literature. Our results encompass 14 existing and newly-developed BERT-based language models across 12 scientific downstream tasks.

We find that model performance on downstream scientific information extraction tasks is not improved significantly or consistently by increasing *any* of the four parameters considered (model size, amount of pretraining data, pretraining time, finetuning time). We attribute these results to both the power of finetuning and limitations in the evaluation datasets, as well as (for the SCHOLARBERT models) small model sizes relative to the large pretraining corpus.

We make the ScholarBERT models available on HuggingFace (https://huggingface.co/globuslabs). While we cannot share the full Public Resource dataset, we have provided a sample of open-access articles from the dataset (https://github.com/tuhz/PublicResourceDatasetSample) in both the original PDF and extracted txt formats to illustrate the quality of the PDF-to-text preprocessing.

Limitations

Our 12 labeled test datasets are from just five domains (plus two multi-disciplinary); five of the 12 are from biomedicine. This imbalance, which reflects the varied adoption of NLP methods across domains, means that our evaluation dataset is necessarily limited. Our largest model, with 770M parameters, may not be sufficiently large to demonstrate scaling laws for language models. We also aim to extend our experiments to tasks other than NER, relation extraction, and text classification, such as question-answering and textual entailment in scientific domains.

References

Jameel Ahmad. 2012. Stylistic features of scientific English: A study of scientific research articles. *English Language and Literature Studies*, 2(1).

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Conference on Empirical Methods in Natural Language Processing*, pages 3615–3620. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Arman Cohan, Waleed Ammar, Madeleine Van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 3586–3596. Association for Computational Linguistics.

Coleridge Initiative. 2020. `https://www.kaggle.com/c/coleridgeinitiative-show-us-the-data`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10.

Maël Fabien, Esaú Villatoro-Tello, Petr Motlicek, and Shantipriya Parida. 2020. Bertaa: Bert fine-tuning for authorship attribution. In *17th International Conference on Natural Language Processing*, pages 127–137. Association for Computational Linguistics.

GROBID. 2008–2022. GROBID. `https://github.com/kermitt2/grobid`.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals Jack W. Rae, and Laurent Sifre. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Shu Huang and Jacqueline M Cole. 2022. BatteryBERT: A pretrained language model for battery database enhancement. *Journal of Chemical Information and Modeling*.

HuggingFace. 2020. English wikipedia corpus. `https://huggingface.co/datasets/wikipedia`. [Online; accessed 08-January-2022].

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 70–75.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, Roger A Sayle, Riza Theresa Batista-Navarro, Rafal Rak, Torsten Huber, Tim Rocktäschel, Sérgio Matos, David Campos, Buzhou Tang, Hua Xu, Tsendsuren Munkhdalai, Keun Ho Ryu, SV Ramanan, Senthil Nathan, Slavko Žitnik, Marko Bajec, Lutz Weber, Matthias Irmer, Saber A Akhondi, Jan A Kors, Shuo Xu, Xin An, Utpal Kumar Sikdar, Asif Ekbal, Masaharu Yoshioka, Thaer M Dieb, Miji Choi, Karin Verspoor, Madian Khabsa, C Lee Giles, Hongfang Liu, Komandur Elayavilli Ravikumar, Andre Lamurias, Francisco M Couto, Hong-Jie Dai, Richard Tzong-Han Tsai, Caglar Ata, Tolga Can, Anabel Usié, Rui Alves, Isabel Segura-Bedmar, Paloma Martínez, Julen Oyarzabal, and Alfonso Valencia. 2015. The CHEMDNER corpus of chemicals and drugs and its annotation principles. *Journal of Cheminformatics*, 7(1):1–17.

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.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: A resource for chemical disease relation extraction. *Database*, 2016.

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. *arXiv preprint arXiv:1907.11692*.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232. Association for Computational Linguistics.

NVIDIA. 2017. NVIDIA Apex (a PyTorch extension). `https://github.com/NVIDIA/apex`.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *18th BioNLP Workshop and Shared Task*, pages 58–65. Association for Computational Linguistics.

Erik F Sang and Fien De Meulder. 2003. CoNLL eval script. `https://www.clips.uantwerpen.be/conll2000/chunking/output.html`.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Hsu, and Kuansan Wang. 2015. An overview of Microsoft Academic Service (MAS) and applications. In *24th International Conference on World Wide Web*, pages 243–246.

Hannah Smith, Zeyu Zhang, John Culnan, and Peter Jansen. 2019. ScienceExamCER: A high-density fine-grained science-domain corpus for common entity recognition. In *12th Language Resources and Evaluation Conference*, pages 4529–4546. European Language Resources Association.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? In *China national conference on Chinese computational linguistics*, pages 194–206. Springer.

Misha Teplitskiy, Eamon Duede, Michael Menietti, and Karim R Lakhani. 2022. How status of research papers affects the way they are read and cited. *Research Policy*, 51(4):104484.

Amalie Trewartha, Nicholas Walker, Haoyan Huo, Sanghoon Lee, Kevin Cruse, John Dagdelen, Alexander Dunn, Kristin A. Persson, Gerbrand Ceder, and Anubhav Jain. 2022. Quantifying the advantage of domain-specific pre-training on named entity recognition tasks in materials science. *Patterns*, 3(4):100488.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *IEEE International Conference on Computer Vision*, pages 19–27.

A Extant BERT-based models

Devlin et al. (2019) introduced BERT-Base and BERT-Large, with ∼110M and ∼340M parameters, as transformer-based masked language models conditioned on both the left and right contexts. Both are pretrained on the English Wikipedia + BooksCorpus datasets.

SciBERT (Beltagy et al., 2019) follows the BERT-Base architecture and is pretrained on data from two domains, namely, biomedical science and computer science. SciBERT outperforms BERT-Base on finetuning tasks by an average of 1.66% and 3.55% on biomedical tasks and computer science tasks, respectively.

BioBERT (Lee et al., 2020) is a BERT-Base model with a pretraining corpus from PubMed abstracts and full-text PubMedCentral articles. Compared to BERT-Base, BioBERT achieves improvements of 0.62%, 2.80%, and 12.24% on biomedical NER, biomedical relation extraction, and biomedical question answering, respectively.

PubMedBERT (Gu et al., 2021), another BERT-Base model targeting the biomedical domain, is also pretrained on PubMed and PubMedCentral text. However, unlike BioBERT, PubMedBERT is trained as a new BERT-Base model, using text drawn exclusively from PubMed and PubMedCentral. As a result, the vocabulary used in PubMedBERT varies significantly from that used in BERT and BioBERT. Its pretraining corpus contains 3.1B words from PubMed abstracts and 13.7B words from PubMedCentral articles. PubMedBERT achieves state-of-the-art performance on the Biomedical Language Understanding and Reasoning Benchmark, outperforming BERT-Base by 1.16% (Gu et al., 2021).

MatBERT (Trewartha et al., 2022) is a materials science-specific model pretrained on 2M journal articles (8.8B tokens). It consistently outperforms BERT-Base and SciBERT in recognizing materials science entities related to solid states, doped materials, and gold nanoparticles, with ∼10% increase in F1 score compared to BERT-Base, and a 1% to 2% improvement compared to SciBERT.

BatteryBERT (Huang and Cole, 2022) is a model pretrained on 400 366 battery-related publications (5.2B tokens). BatteryBERT has been shown to outperform BERT-Base by less than 1% on the SQuAD question answering task. For battery-specific question-answering tasks, its F1 score is around 5% higher than that of BERT-base.

B ScholarBERT Pretraining Details

B.1 Tokenization

The vocabularies generated for PRD_1 and PRD_10 differed only in 1–2% of the tokens; however, in an initial study, the PRD_100 vocabulary differed from that of PRD_10 by 15%. A manual inspection of the PRD_100 vocabulary revealed that many common English words such as “is,” “for,” and “the” were missing. We determined that these omissions were an artifact of PRD_100 being sufficiently large to cause integer overflows in the unsigned 32-bit-integer token frequency counts used by HuggingFace’s tokenizers library. For example, “the” was not in the final vocabulary because the token “th” overflowed. Because WordPiece iteratively merges smaller tokens to create larger ones, the absence of tokens like “th” or “##he” means that “the” could not appear in the final vocabulary.

We modified the tokenizers library to use unsigned 64-bit integers for all frequency counts, and recreated a correct vocabulary for PRD_100. Interestingly, models trained on the PRD_100 subset with the incorrect and correct vocabularies exhibited comparable performance on downstream tasks.

B.2 RoBERTa Optimizations

RoBERTa introduces many optimizations for improving BERT pretraining performance (Liu et al., 2019). 1) It uses a single phase training approach whereby all training is performed with a maximum sequence length of 512. 2) Unlike BERT which randomly introduces a small percentage of shortened sequence lengths into the training data, RoBERTa does not randomly use shortened sequences. 3) RoBERTa uses dynamic masking, meaning that each time a batch of training samples is selected at runtime, a new random set of masked tokens is selected; in contrast, BERT uses static masking, pre-masking the training samples prior to training. BERT duplicates the training data 10 times each with a different random, static masking. 4) RoBERTa does not perform Next Sentence Prediction during training. 5) RoBERTa takes sentences contiguously from one or more documents until the maximum sequence length is met. 6) RoBERTa uses a larger batch size of 8192. 7) RoBERTa uses byte-pair encoding (BPE) rather than WordPiece. 8) RoBERTa uses an increased vocabulary size of 50 000, 67% larger than BERT. 9) RoBERTa trains for more iterations (up to 500 000) than does BERT-Base (31 000).

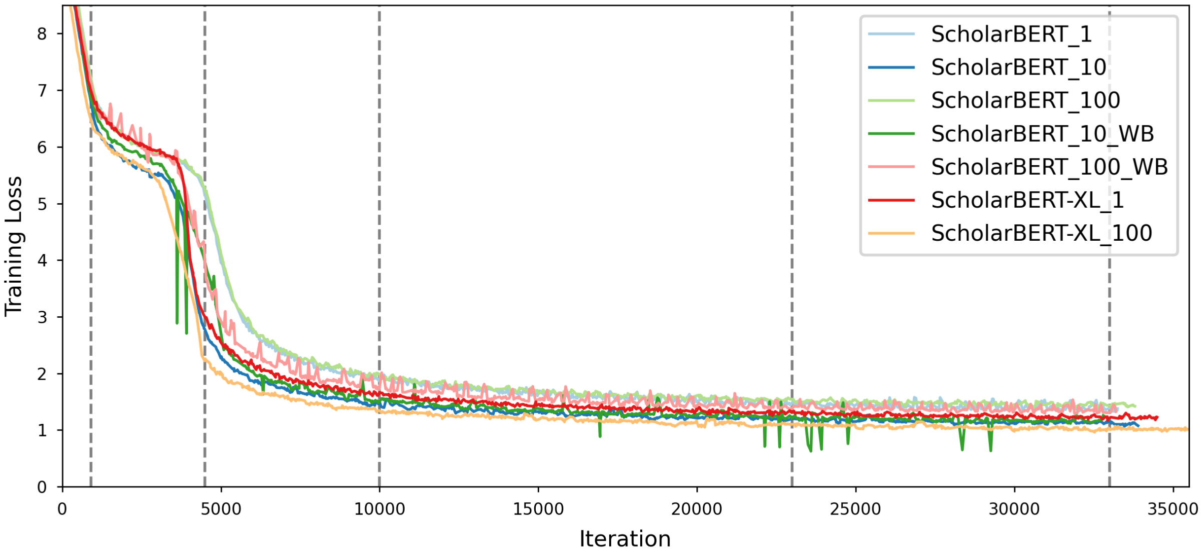

Figure 1: Pretraining loss plots for the SCHOLARBERT models listed in Table 1. The vertical dashed lines indicate the approximate locations of the iteration checkpoints selected for evaluation in Section 4.1.

Name	Description	Domain	Tokens
Wiki	English-language Wikipedia articles (HuggingFace, 2020)	Gen	2.5B
Books	BookCorpus (Zhu et al., 2015; HuggingFace, 2020): Full text of 11038 books	Gen	0.8B
SemSchol	1.14M papers from Semantic Scholar (Cohan et al., 2019), 18% in CS, 82% in Bio	Bio, CS	3.1B
$\texttt{PubMed}_A$	Biomedical abstracts sampled from PubMed (Gu et al., 2021)	Bio	3.1B
$\texttt{PubMed}_B$	Biomedical abstracts sampled from PubMed (Lee et al., 2020)	Bio	4.5B
PMC	Full-text biomedical articles sampled from PubMedCentral (Gu et al., 2021)	Bio	13.7B
MatSci	2M peer-reviewed materials science journal articles (Trewartha et al., 2022)	Materials	8.8B
Battery	0.4M battery-related publications (Huang and Cole, 2022)	Materials	5.2B
PRD_1	1% of the English-language research articles from the Public Resource dataset	Sci	2.2B
PRD_10	10% of the English-language research articles from the Public Resource dataset	Sci	22B
PRD_100	100% of the English-language research articles from the Public Resource dataset	Sci	221B

Table 4: Pretraining corpora used by models in this study. The domains are Bio=biomedicine, CS=computer science, Gen=general, Materials=materials science and engineering and Sci=broad scientific.

We adopt RoBERTa training methods, with three key exceptions. 1) Unlike RoBERTa, we randomly introduce smaller length samples because many of our downstream tasks use sequence lengths much smaller than the maximum sequence length of 512 that we pretrain with. 2) We pack training samples with sentences drawn from a single document, as the RoBERTa authors note that this results in slightly better performance. 3) We use WordPiece encoding rather than BPE, as the RoBERTa authors note that BPE can result in slightly worse downstream performance.

Hyperparameter	Value
Steps	33 000
Optimizer	LAMB
LR	0.0004
LR Decay	Linear
LR Warmup Steps	0.06%
Batch Size	32 768
Precision	FP16
Weight Decay	0.01
Attention Dropout	10%
Hidden Dropout	10%
Hidden Activation	GELU

Table 5: Pretraining hyperparameters. All SCHOLARBERT variants use the same pretraining hyperparameters.

B.3 Hardware and Software Stack

We perform data-parallel pretraining on a cluster with 24 nodes, each containing eight 40 GB NVIDIA A100 GPUs. In data-parallel distributed training, a copy of the model is replicated on each GPU, and, in each iteration, each GPU computes on a unique local mini-batch. At the end of the iteration, the local gradients of each model replica are averaged to keep each model replica in sync. We perform data-parallel training of SCHOLARBERT models using PyTorch's distributed data-parallel model wrapper and 16 A100 GPUs. For the larger SCHOLARBERT-XL models, we use the Deep-

Speed data-parallel model wrapper and 32 A100 GPUs. The DeepSpeed library incorporates a number of optimizations that improve training time and reduced memory usage, enabling us to train the larger model in roughly the same amount of time as the smaller model.

We train in FP16 with a batch size of 32 768 for ∼33 000 iterations (Table 5). To achieve training with larger batch sizes, we employ NVIDIA Apex's FusedLAMB (NVIDIA, 2017) optimizer, with an initial learning rate of 0.0004. The learning rate is warmed up for the first 6% of iterations and then linearly decayed for the remaining iterations. We use the same masked token percentages as are used for BERT. Training each model requires roughly 1000 node-hours, or 8000 GPU-hours.

Figure 1 depicts the pretraining loss for each SCHOLARBERT model. We train each model past the point of convergence and take checkpoints throughout training to evaluate model performance as a function of training time.

C Evaluation Tasks

We evaluate the models on eight NER tasks and four sentence-level tasks. For the NER tasks, we use eight annotated scientific NER datasets:

1. BC5CDR (Li et al., 2016): An NER dataset identifying diseases, chemicals, and their interactions, generated from the abstracts of 1500 PubMed articles containing 4409 annotated chemicals, 5818 diseases, and 3116 chemical-disease interactions, totaling 6283 unique entities.

2. JNLPBA (Kim et al., 2004): A bio-entity recognition dataset of molecular biology concepts from 2404 MEDLINE abstracts, consisting of 21 800 unique entities.

3. SciERC (Luan et al., 2018): A dataset annotating entities, relations, and coreference clusters in 500 abstracts from 12 AI conference/workshop proceedings. It contains 5714 distinct named entities.

4. NCBI-Disease (Doğan et al., 2014): Annotations for 793 PubMed abstracts: 6893 disease mentions, of which 2134 are unique.

5. ChemDNER (Krallinger et al., 2015): A chemical entity recognition dataset derived from 10 000 abstracts containing 19 980 unique chemical entity mentions.

6. MatSciNER (Trewartha et al., 2022): 800 annotated abstracts from solid state materials publications sourced via Elsevier's Scopus/-ScienceDirect, Springer-Nature, Royal Society of Chemistry, and Electrochemical Society. Seven types of entities are labeled: inorganic materials (MAT), symmetry/phase labels (SPL), sample descriptors (DSC), material properties (PRO), material applications (APL), synthesis methods (SMT), and characterization methods (CMT).

7. ScienceExam (Smith et al., 2019): 133K entities from the Aristo Reasoning Challenge Corpus of 3rd to 9th grade science exam questions.

8. Coleridge (Coleridge Initiative, 2020): 13 588 entities from sociology articles indexed by the Inter-university Consortium for Political and Social Research (ICPSR).

The sentence-level downstream tasks are relation extraction on the ChemProt (biology) and SciERC (computer science) datasets, and sentence classification on the Paper Field (multidisciplinary) and Battery (materials) dataset:

1. ChemProt consists of 1820 PubMed abstracts with chemical-protein interactions annotated by domain experts (Peng et al., 2019).

2. SciERC, introduced above, provides 4716 relations (Luan et al., 2018).

3. The Paper Field dataset (Beltagy et al., 2019), built from the Microsoft Academic Graph (Sinha et al., 2015), maps paper titles to one of seven fields of study (geography, politics, economics, business, sociology, medicine, and psychology), with each field of study having around 12K training examples.

4. The Battery Document Classification dataset (Huang and Cole, 2022) includes 46 663 paper abstracts, of which 29 472 are labeled as battery and the other 17 191 as non-battery. The labeling is performed in a semi-automated manner. Abstracts are selected from 14 battery journals and 1044 non-battery journals, with the former labeled "battery" and the latter "non-battery."

D Extended Results

Table 6 shows average F1 scores with standard deviations for the NER tasks, each computed over five runs; Figure 2 presents the same data, with standard deviations represented by error bars. Table 7 and Figure 3 show the same for sentence classification tasks. The significant overlaps of error bars for NCBI-Disease, SciERC NER, Coleridge, SciERC Sentence Classification, and ChemProt corroborate our observation in Section 4 that on-domain pretraining provides only marginal advantage for downstream prediction over pretraining on a different domain or a general corpus.

	BC5CDR	**JNLPBA**	**NCBI-Disease**	**SciERC**
BERT-Base	85.36 ± 0.189	72.15 ± 0.118	84.28 ± 0.388	56.73 ± 0.716
BERT-Large	86.86 ± 0.321	72.80 ± 0.299	84.91 ± 0.229	59.20 ± 1.260
SciBERT	88.43 ± 0.112	73.24 ± 0.184	86.95 ± 0.714	59.36 ± 0.390
PubMedBERT	89.34 ± 0.185	74.53 ± 0.220	87.91 ± 0.267	59.03 ± 0.688
BioBERT	88.01 ± 0.133	73.09 ± 0.230	87.84 ± 0.513	58.24 ± 0.631
MatBERT	86.44 ± 0.156	72.56 ± 0.162	84.94 ± 0.504	58.52 ± 0.933
BatteryBERT	87.42 ± 0.308	72.78 ± 0.190	87.04 ± 0.553	59.00 ± 1.174
SB_1	87.27 ± 0.189	73.06 ± 0.265	85.49 ± 0.998	58.62 ± 0.602
SB_10	87.69 ± 0.433	73.03 ± 0.187	85.65 ± 0.544	58.39 ± 1.643
SB_100	87.84 ± 0.329	73.47 ± 0.210	85.92 ± 1.040	58.37 ± 1.845
SB_10_WB	86.68 ± 0.397	72.67 ± 0.329	84.51 ± 0.838	57.34 ± 1.199
SB_100_WB	86.89 ± 0.543	73.16 ± 0.211	84.88 ± 0.729	58.43 ± 0.881
SB-XL_1	87.09 ± 0.179	73.14 ± 0.352	84.61 ± 0.730	58.45 ± 1.614
SB-XL_100	87.46 ± 0.142	73.25 ± 0.300	84.73 ± 0.817	57.26 ± 2.146

	ChemDNER	**MatSciNER**	**ScienceExam**	**Coleridge**
BERT-Base	84.84 ± 0.004	78.51 ± 0.300	78.37 ± 0.004	57.75 ± 1.230
BERT-Large	85.83 ± 0.022	82.16 ± 0.040	82.32 ± 0.072	57.46 ± 0.818
SciBERT	85.76 ± 0.089	82.64 ± 0.054	78.83 ± 0.004	54.07 ± 0.930
PubMedBERT	87.96 ± 0.094	82.63 ± 0.045	69.73 ± 0.872	57.71 ± 0.107
BioBERT	85.53 ± 0.130	81.76 ± 0.094	78.60 ± 0.072	57.04 ± 0.868
MatBERT	86.09 ± 0.170	83.35 ± 0.085	80.01 ± 0.027	56.91 ± 0.434
BatteryBERT	86.49 ± 0.085	82.94 ± 0.309	78.14 ± 0.103	59.87 ± 0.398
SB_1	85.25 ± 0.063	80.87 ± 0.282	82.75 ± 0.049	55.34 ± 0.742
SB_10	85.80 ± 0.094	80.61 ± 0.747	83.24 ± 0.063	53.41 ± 0.380
SB_100	85.90 ± 0.063	82.09 ± 0.022	83.12 ± 0.085	54.93 ± 0.063
SB_10_WB	83.94 ± 0.058	78.98 ± 1.190	83.00 ± 0.250	54.29 ± 0.080
SB_100_WB	84.31 ± 0.080	80.84 ± 0.161	82.43 ± 0.031	54.00 ± 0.425
SB-XL_1	85.81 ± 0.054	82.84 ± 0.228	81.09 ± 0.170	55.94 ± 0.899
SB-XL_100	85.73 ± 0.058	81.75 ± 0.367	80.72 ± 0.174	54.54 ± 0.389

Table 6: NER F1 scores for each of 14 models (rows), when the model is finetuned on eight different domain datasets and the resulting finetuned model applied to that dataset's associated NER task (columns). In each case, we give the average value and its standard deviation over five runs.

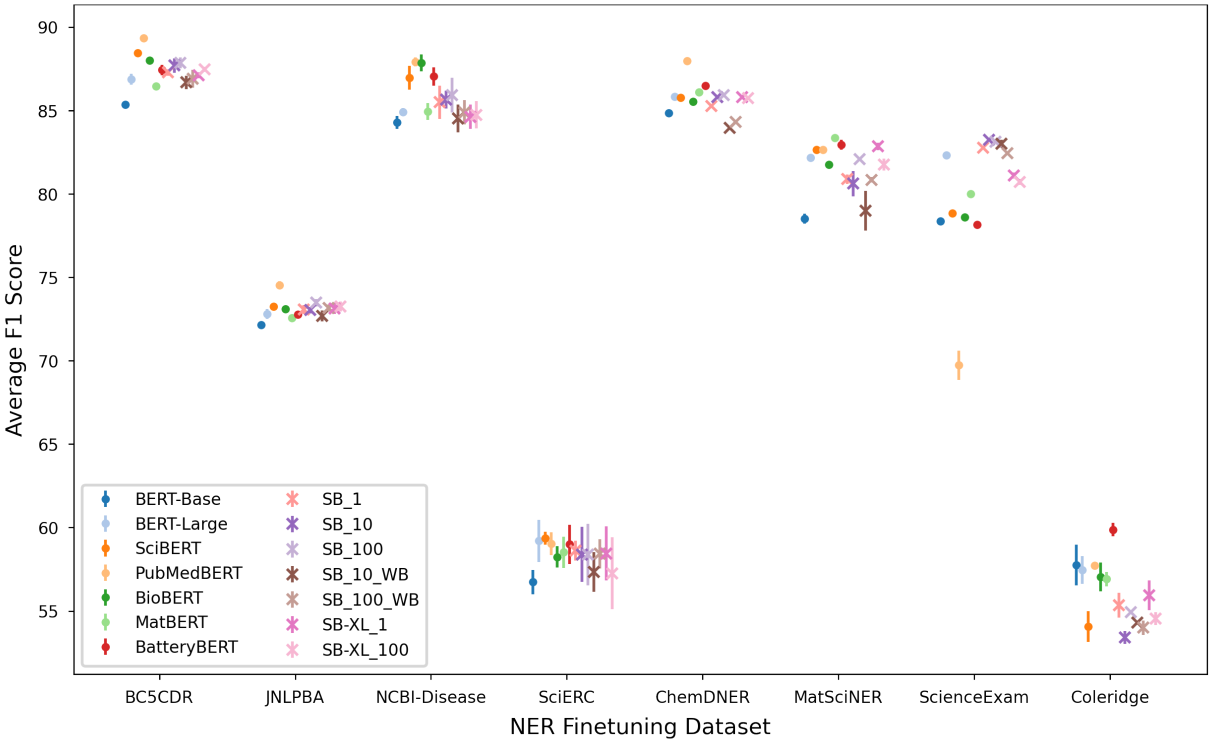

Figure 2: NER F1 scores from Table 6, with standard deviations represented by error bars.

	SciERC	ChemProt	PaperField	Battery
BERT-Base	74.95 ± 1.596	83.70 ± 0.472	72.83 ± 0.082	96.31 ± 0.087
BERT-Large	80.14 ± 2.266	88.06 ± 0.353	73.12 ± 0.125	96.90 ± 0.156
SciBERT	79.26 ± 0.498	89.80 ± 0.263	73.19 ± 0.046	96.38 ± 0.153
PubMedBERT	77.45 ± 0.964	91.78 ± 0.096	73.93 ± 0.099	96.58 ± 0.148
BioBERT	80.12 ± 0.179	89.27 ± 0.281	73.07 ± 0.074	96.06 ± 0.200
MatBERT	79.85 ± 0.121	88.15 ± 0.026	71.50 ± 0.135	96.33 ± 0.106
BatteryBERT	78.14 ± 0.550	88.33 ± 0.939	73.28 ± 0.022	96.06 ± 0.437
SB_1	73.01 ± 0.248	83.04 ± 0.150	72.77 ± 0.060	94.67 ± 0.671
SB_10	75.95 ± 0.203	82.92 ± 0.792	72.94 ± 0.182	92.83 ± 3.758
SB_100	76.19 ± 1.592	87.60 ± 0.324	73.14 ± 0.085	92.38 ± 5.789
SB_10_WB	73.17 ± 1.254	81.48 ± 1.705	72.37 ± 0.115	93.15 ± 1.763
SB_100_WB	76.71 ± 2.114	83.98 ± 0.252	72.29 ± 0.048	95.55 ± 0.272
SB-XL_1	74.85 ± 1.497	90.60 ± 0.246	73.22 ± 0.009	88.75 ± 4.035
SB-XL_100	80.99 ± 0.900	89.18 ± 0.499	73.66 ± 0.113	95.44 ± 0.100

Table 7: Sentence classification F1 scores for each of 14 models (rows), when the model is finetuned on one of four different domain datasets and the finetuned model is applied to that dataset's associated sentence classification task (columns). In each case, we give the average value and its standard deviation over five runs.

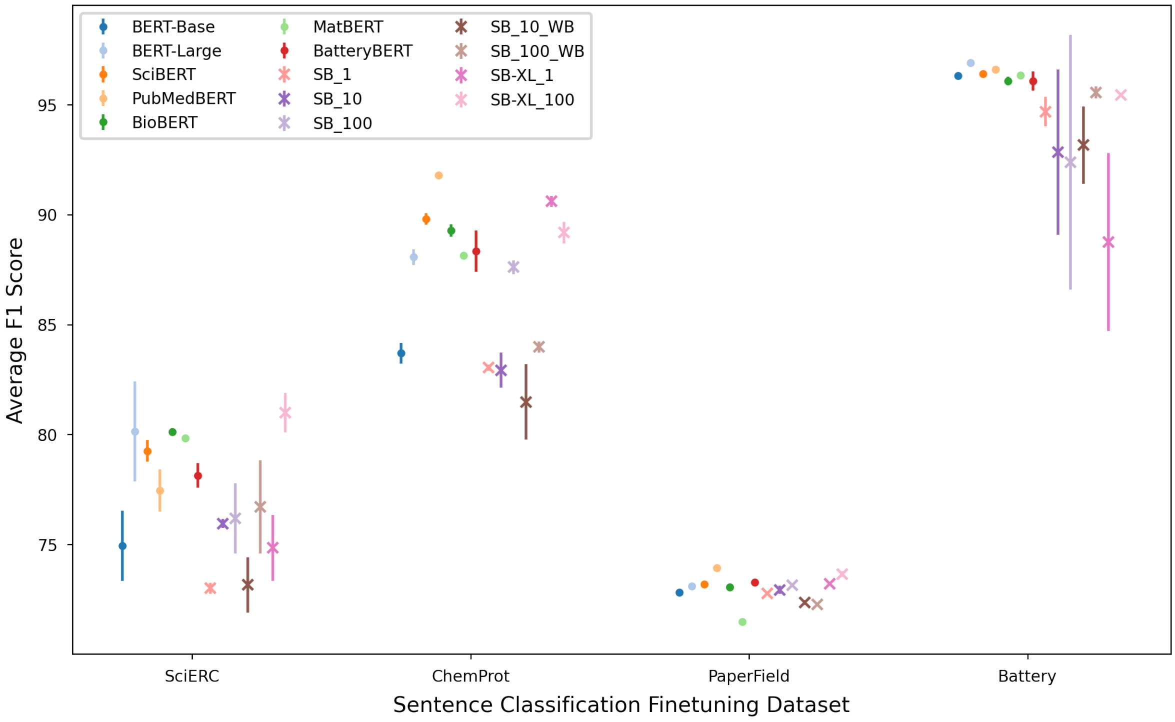

Figure 3: Sentence classification F1 scores from Table 7, with standard deviations represented by error bars.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section Limitations

☒ A2. Did you discuss any potential risks of your work?
Our article reports findings on evaluating language models on scientific information extraction tasks, which we do not believe could pose any risk.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section Abstract and Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ 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)?
No response.

□ 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?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ 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.
No response.

C ☑ Did you run computational experiments?

Section 4

☑ 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.3 Hardware and Software Stack

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?
Section 3 Table 1 and Appendix B Table 5

☑ 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?
Section 4 and Appendix D

☑ 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.3 Hardware and Software Stack

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.?
No response.

□ 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)?
No response.

□ 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?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.